
Article:
Ye, Fei and Bors, Adrian Gheorghe orcid.org/0000-0001-7838-0021 (2021) Lifelong Teacher-Student Network Learning. IEEE Transactions on Pattern Analysis and Machine Intelligence. pp. 6280-6296. ISSN 0162-8828

https://doi.org/10.1109/TPAMI.2021.3092677

Lifelong Teacher-Student Network Learning

Fei Ye and Adrian G. Bors, *Senior Member, IEEE*
Department of Computer Science, University of York, York YO10 5GH, UK
E-mail: fy689@york.ac.uk, adrian.bors@york.ac.uk

Abstract—A unique cognitive capability of humans consists in their ability to acquire new knowledge and skills from a sequence of experiences. Meanwhile, artificial intelligence systems are good at learning only the last given task without being able to remember the databases learnt in the past. We propose a novel lifelong learning methodology by employing a Teacher-Student network framework. While the Student module is trained with a new given database, the Teacher module would remind the Student about the information learnt in the past. The Teacher, implemented by a Generative Adversarial Network (GAN), is trained to preserve and replay past knowledge corresponding to the probabilistic representations of previously learn databases. Meanwhile, the Student module is implemented by a Variational Autoencoder (VAE) which infers its latent variable representation from both the output of the Teacher module as well as from the newly available database. Moreover, the Student module is trained to capture both continuous and discrete underlying data representations across different domains. The proposed lifelong learning framework is applied in supervised, semi-supervised and unsupervised training. The code is available :
https://github.com/dtuzi123/Lifelong-Teacher-Student-Network-Learning

Index Terms—Lifelong representation Learning, Variational Autoencoders, Generative Adversarial Nets, Teacher -Student framework.

1 Introduction

Humans have an inherent ability to memorize, interpret and transfer knowledge across tasks, [1]. Lifelong learning represents the capability of people or animals of being able to continually acquire new skills or novel knowledge from a sequence of tasks while also maintaining their performance on previously learnt tasks [2]. When presented with a new task, humans would use their previously learnt experience in order to understand it. The more related two tasks are, the easiest is to learn them one after the other. This ability is essential for adaptation and solving many real-world problems and would be very useful if it could be implemented in artificial systems in order to advance their capabilities. Artificial learning systems, able to learn new information from multiple sources while expanding their already assimilated cognitive abilities, would be able to solve multiple challenges [3]. However, lifelong learning remains a serious challenge for deep learning applications. While deep learning approaches perform well in many specific data classification applications [4], they suffer from the catastrophic forgetting problem [5], [6], [7], [8] when attempting to learn new tasks. This happens because a deep learning model, which had been trained initially on a specific database, loses that knowledge when is trained for a new task on a novel data set.

A pre-trained machine learning system can be used on a specific target domain by either using transfer learning [9] or domain adaptation [10], [11]. While the former situation assumes that domains differ in the sample space, in the latter case the data distributions could change between the datasets. The challenge in this case is to overcome the differences between the domains in order to ensure a good generalization, [12].

Prior research aiming to alleviate catastrophic forgetting was often focused on regularization and using dynamic architectures. For instance, regularization based approaches would normally impose a larger penalty for changing the model parameters in order to relieve catastrophic forgetting [13]. However, these approaches do not work well when learning entirely different data sets. Dynamic architecture approaches would either freeze the weights for sections of the network or add new processing nodes when learning new tasks. The drawback for these approaches is that they invariably require additional network structures, thus increasing the number of parameters requiring training for storing additional information.

Can we train a single model able to capture meaningful representations across multiple domains through sequential learning? Variational Autoencoders (VAEs), such as β-VAE [4], or Generative Adversarial Networks (GANs), such as InfoGAN [14] have been used to learn disentangled representations. VAE based approaches normally would modify the main objective function by imposing a larger penalty on the Kullback-Leibler (KL) divergence between the prior and posterior distributions in order to encourage disentanglement on the latent variables, [4]. In other approaches, the total correlation is used as a regularization term in the objective function for ensuring the disentanglement among categories of characteristics in the feature space [15], [16], [17]. InfoGAN [14], learns an interpretable subset of codes by maximizing the mutual information between the latent variables and the generation process. Nevertheless, these approaches only perform well on independent and identically distributed data drawn from the same probabilistic representation [18]. Learning disentangled representations within the lifelong learning setting is challenging given that the previously learnt experiences will be quickly forgotten when training on a new domain.

This research study proposes a Lifelong learning Teacher-Student (LTS) framework, which brings the following contributions :

1) The Teacher-Student Lifelong learning forms an artificial symbiosis system of two networks: Teacher and Student. The Teacher component is implemented by a powerful data generator network such as a GAN, while the Student is implemented by a latent representation generative model. The proposed model can overcome forgetting while learning probabilistic data representations over time.
2) We use conditional priors for encouraging the information learnt from different domains to have different posteriors, resulting in a better inference across domains during the lifelong learning.
3) The LTS framework learns meaningful representations across domains by employing a disentangled representation methodology.
4) The proposed model is adapted to be used in supervised, semi-supervised and unsupervised lifelong learning.

Related research into lifelong learning is presented in Section 2. The LTS system is described in Section 3, while its training is outlined in Section 4. The application of the proposed model for lifelong learning in semi-supervised and unsupervised applications is provided in Section 5. The error bounds for the lifelong learning of the Student module are derived in Section 6. Experimental results are provided in Section 7, while the conclusions are drawn in Section 8.

2 Related works

In this section, we review prior research studies on lifelong learning.

2.1 Lifelong learning

Lifelong learning remains a challenging task for machine learning [19]. A classifier trained under the lifelong setting, aiming to learn sequentially probabilistic representations of several databases, suffers from catastrophic forgetting [13]. This happens due to the fact that previously learnt knowledge is overwritten when learning a new task, through changing network parameter values. Existing lifelong learning approaches can be divided into three categories: regularization, dynamic architectures and memory replay.

Regularization. Regularization approaches normally impose constraints on the objective function during training in order to alleviate catastrophic forgetting. Changes in the weights of the neural network are penalized by considering a regularization term in the objective function. For instance, Li *et al.* [13] introduced a lifelong learning system, called Learning without Forgetting (LwF), which encourages the predictions for each data sample to be similar to the outputs from the original network by using Knowledge Distillation [20]. A similar approach is called Less-Forgetting Learning [21], which aims to preserve the performance of the network for old tasks by learning a shared representation across multiple tasks. This approach assumes that the final decision layer for each task should not change too much. Kirkpatrick *et al.* [22] introduced the Elastic Weight Consolidation (EWC) algorithm which encourages the weights of a neural network deemed significant to be close to their previous values when learning a new task. Zenke *et al.* [23] proposed a lifelong learning algorithm to alleviate catastrophic forgetting by imposing a penalty on the changes of important weights when learning each task. Reducing significant changes in the weights can lead to the preservation of the network performance in the previously learnt tasks. Ensemble-based methods [6], [7], [24] have also been used to deal with catastrophic forgetting. These approaches normally train multiple classifiers and then combine their predictions.

Dynamic architectures. These approaches use a flexible network architecture, which can be dynamically changed when learning new tasks. Resu *et al.* [25] proposed the Progressive Neural Network which starts with a basic structure and increases its complexity when training with new information. In order to avoid catastrophic forgetting, this approach considers sub-networks, for each learnt task, whose parameters are frozen when learning new tasks. Zhou *et al.* [26] introduced an incremental feature learning algorithm. This approach adds features learnt from new data sets while ensuring a compact feature representation, through merging whenever necessary and preventing over-fitting. Cortes *et al.* [27] proposed an adaptive learning algorithm called AdaNet, which jointly adapts the network architecture and ensures a trade-off between the empirical risk minimization and model complexity. Xiao *et al.* [28] proposed a learning algorithm which increases hierarchically the capacity of a neural network, while Part *et al.* [29] combined a pre-trained convolution neural network (CNN) and a self-organizing incremental neural network (SOINN). The pre-trained CNN provides good representations from the previously learnt data sets, while the topology of SOINN is evolving continuously according to the input data distribution.

Memory replay. Typical approaches for memory replay are using generative models such as Generative Adversarial Networks (GAN) or Variational Autoencoders (VAE). A GAN consists of a generator network G and a discriminator network D performing a two-player MiniMax game, where G aims to produce realistic data which would aim to fool D into believing they are real data, while the latter aims to distinguish such fake data from the real. VAEs [30], [31] represent a probabilistic graphical model which consists of two components: the encoder network which models a representative variable latent space for the data while the decoder is trained to recover the real data from the latent variable space and implements an inverse mapping of the encoder. The learning goal of VAEs consists of maximizing the log-likelihood of data reconstruction while minimizing the Kullback-Leibler (KL) divergence between the latent variable variational approximation and the prior.

Shin *et al.* [32], proposed a dual-model architecture based on a deep generative model and a classifier. This computational framework replays past knowledge by generating pseudo-data using the generative model trained on previous tasks. The information associated with a new task is interleaved with generated data, and used together to train the task solver. However, this approach would consider only classification tasks and is unable to learn any meaningful latent data representations due to lacking an inference mechanism. Ramapuram *et al.* [33] proposed the Lifelong Generative Modeling (LGM) which employs VAEs for two networks working in tandem: a Teacher and a

Student. During the training past knowledge is replayed by the Teacher network whose decoder maps latent variables, sampled from the prior distribution, into the data space. Achille *et al.* [18] introduced a VAE based lifelong generative model for disentangled representation learning, called VASE, which is able to learn meaningful latent variables across multiple domains. VASE is based on the Minimum Description Length (MDL) principle, which progressively increases the network size in order to accommodate learning new data. MDL represents a trade-off criterion between the size of the network and its learning performance. The quality of the data generated from previously learnt knowledge in algorithms such as LGM [33] or VASE [18] depends on the generative abilities of VAEs, which usually is not great and would result in blurred images. These models do not perform well in the case of complex data due to a rather poor replay of the knowledge from the previously learnt databases. Seff *et al.* [34] proposed the Augmented Generator objective function, based on a GAN, which is known as a better data generator than VAEs. Nevertheless, this model is applied on rather simple data.

3 Lifelong Teacher-Student Network

The standard generative models usually aim to estimate a set of network parameters, maximizing the marginal likelihood for a data set $\mathcal{X}$, drawn from its probabilistic representation $p(\mathcal{X})$. Nevertheless, in real situations, artificial systems would have to learn tasks sequentially, at certain time intervals, from several databases, $\mathcal{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_k\}$. A model training with data from a new database will adapt and change its parameters through training.

3.1 The Lifelong Learning Framework

In this research study, we focus on the lifelong learning problem [3] in which a model is trained to learn a sequence of tasks, each defined by learning a probabilistic representation corresponding to a specific database. During the training, we only acquire the information corresponding to the current task while past data sets are considered as not being available, [8]. A lifelong learning model would require to preserve the information learnt during the previous learning cycles while also learning new tasks using the data from a freshly available database. We propose a novel Teacher-Student framework for lifelong learning, where the Teacher component is designed to remember all past knowledge, while the Student module would be trained using two input sources: the current task, defined by the data contained in the new database $\mathbf{X}_k$, and the information provided by the Teacher module, representing past information. By using the learnt knowledge, the Student module is able to perform specific tasks such as classification or discovering disentangled representations, characteristic to the entire data space $\mathcal{X}$. Existing Teacher-Student networks focus on how to transfer knowledge from a more complex network into a smaller, distilled model, by using compression techniques [20]. While such approaches provide a good performance [35], [36], they are unable to preserve well previously acquired information and related tasks.

Let us consider a model $\mathcal{F}(\mathcal{X})$, which is trained on a sequence of training data sets $\{\mathbf{x}_1 \sim \mathbf{X}_1, \mathbf{x}_2 \sim \mathbf{X}_2, \ldots, \mathbf{x}_k \sim \mathbf{X}_k\}$. Each data set $\mathbf{x}_i$, $i = 1, \ldots, k$ is assumed to be characterized by a distinct distribution $p(\mathbf{x}_i)$. After training, the model $\mathcal{F}(\mathcal{X})$ can make predictions on any of the data sets $\{\mathbf{x}_i \in \mathcal{X} | i = 1, \ldots, k\}$. The lifelong learning in artificial systems implies that the deep learning system learns about the latest k-th given data set $\mathbf{x}_k \sim \mathbf{X}_k$, while none of the previously observed data sets $\{\mathbf{X}_i, i < k\}$ are available. For ensuring addressing the most general situations, in this study we consider three different characteristic latent variables, characterizing each database $\{\mathbf{X}_i | i = 1, \ldots, k\}$, which are inferred by the Teacher-Student network: continuous $\mathbf{z}$, discrete $\mathbf{s}$, and the domain latent δ, variables, respectively. While the discrete variables model data attributes, such as class labels, the continuous latent variables model the variation within the whole latent space. Each component of the domain variables $\delta = \{\delta_i | i = 1, \ldots, k\}$, is a one-hot vector representing identifiers for each database within the lifelong learning process.

Let us consider that $p(\mathbf{x}_k)$ represents the currently available empirical data distribution and $p(\mathbf{x}_1, \ldots, \mathbf{x}_{k-1})$ are the probabilistic representations of the previously learnt data distributions. The proposed lifelong learning is defined as learning a representation model:

$$p(\mathcal{X}) = \iiint p(\mathcal{X}|\mathbf{z}, \mathbf{s}, \delta) p(\mathbf{z}, \mathbf{s}, \delta) d\mathbf{z}\, d\mathbf{s}\, d\delta, \tag{1}$$

where we have continuous latent spaces represented by $\mathbf{z}$, discrete variables $\mathbf{s}$, and the domain δ latent spaces. After dropping $\mathbf{s}$ and δ, for the sake of simplification, we can show how the latent representation is used to model data in probabilistic terms through the Bayes' rule:

$$p(\mathbf{z}|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) \propto p(\widehat{\mathbf{x}}_{\mathbf{k-1}}, \mathbf{x}_\mathbf{k}|\mathbf{z}) \mathbf{p}(\mathbf{z}) \propto \mathbf{p}(\mathbf{z}|\mathbf{x}_\mathbf{k}) \mathbf{p}(\widehat{\mathbf{x}}_{\mathbf{k-1}}|\mathbf{z}) \tag{2}$$

where $p(\mathbf{z}|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k)$ represents the probability of the latent space, estimated by the Student module, defining the entire latent space of the data $\mathcal{X}$, using data sampled directly from the latest available data set, defined by $p(\mathbf{x}_k)$ and the data generated by the Teacher module, $p(\widehat{\mathbf{x}}_{k-1})$, corresponding to the previously learnt knowledge.

3.2 Teacher module

For the Teacher we consider a data generative model such as a GAN model [37]. However, classical GAN networks are well known for their instability, sometimes generating images which are not realistic. Consequently, we consider the Wasserstein GAN (WGAN) [38], which uses the Earth-Mover distance as the optimization function for training. WGAN provides better training stability while the quality of generated images is much better when compared to classical GAN [37].

Let us consider $p(\widehat{\mathbf{x}}_k)$ as the output probability density function of the generator network of the WGAN, $G_{\psi_k}(\mathbf{z}, \delta)$ estimated through adversarial learning from k-th database, where $\mathbf{z}$ is sampled from the Gaussian distribution $\mathcal{N}(0, \mathbf{I})$. When a new task, corresponding to a database $\mathbf{X}_k$, is identified for training, the Teacher module is trained with a mixed data set, corresponding to a joint probability density function $p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k)$. The probability of sampling the data

for the joint distribution depends on the importance of the new task $\mathbf{x}_k \sim \mathbf{X}^k$ when compared to that of the previously learnt tasks, $\widehat{\mathbf{x}}_{k-1} \sim p(\widehat{\mathbf{x}}_{k-1})$.

The following WGAN objective function is considered for the Teacher module:

$$\min_G \max_{D\in\mathcal{A}} \left\{ \mathbb{E}_{\mathbf{x}\sim p(\widehat{\mathbf{x}}_{k-1},\mathbf{x}_k)}[D(\mathbf{x})] - \mathbb{E}_{\widehat{\mathbf{x}}\sim G_{\psi_k}(\mathbf{z},\delta)}[D(\widehat{\mathbf{x}})] \right\}, \tag{3}$$

where D is the decision of the WGAN discriminator, $\mathcal{A}$ represents a set of 1-Lipschitz functions, with $\|D(\mathbf{x})\|_L \leq 1$ in order to avoid the mode collapse, which is typical in classical GANs, $\mathbf{x} \sim p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k)$, where the data used for training the WGAN network is sampled in equal probability ratios from $p(\widehat{\mathbf{x}}_{k-1})$, representing the data generated after learning the previous database $\mathbf{X}_{k-1}$, and data sampled from $p(\mathbf{x}_k)$, corresponding to the new database $\mathbf{X}_k$. Meanwhile, $\widehat{\mathbf{x}} \sim G_{\psi_k}(\mathbf{z}, \delta)$ represents the data generated by the generator G, defined by the parameters ψ_k characterized by the random continuous variable $\mathbf{z}$ and discrete variables δ. For the domain probability density function $p(\delta)$ we consider a categorical distribution $Cat(\varsigma_1, \ldots, \varsigma_k)$ where ς_i is the probability of observing i-th task associated with the corresponding database, $i = 1, \ldots, k$. The domain variable δ would encode information characteristic to a specific task acquired during the lifelong learning.

Observation 1. The Teacher network represents the probabilistic storage container for the entire previously learnt knowledge by the Teacher-Student network. The probability density of generated data by the Teacher module represents statistical correlations of the data from all taught tasks.

Proof. We can describe the probability of the data generated by the Teacher module when learning the k-th task, $p(\widehat{\mathbf{x}}_k)$ as depending on the probability of the data generated by the Teacher after learning the $k-1$-th task, $p(\widehat{\mathbf{x}}_{k-1})$ and the probability describing the new database $p(\mathbf{x}_k)$, as :

$$\begin{aligned} p(\widehat{\mathbf{x}}_k) &= \iint p(\widehat{\mathbf{x}}_k|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) d\widehat{\mathbf{x}}_{k-1} d\mathbf{x}_k \\ &= \iint p(\widehat{\mathbf{x}}_k|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) p(\widehat{\mathbf{x}}_{k-1}) p(\mathbf{x}_k) d\widehat{\mathbf{x}}_{k-1} d\mathbf{x}_k. \end{aligned} \tag{4}$$

After using mathematical induction for describing the recursive learning of several databases during the lifelong learning, while considering the data generation by the Teacher network, we have :

$$\begin{aligned} p(\widehat{\mathbf{x}}_k) &= \iiiint p(\widehat{\mathbf{x}}_{k-1}|\widehat{\mathbf{x}}_{k-2}, \mathbf{x}_{k-1}) p(\widehat{\mathbf{x}}_{k-2}) p(\mathbf{x}_{k-1}) \\ &\quad p(\widehat{\mathbf{x}}_k|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) p(\mathbf{x}_k) d\widehat{\mathbf{x}}_{k-2} d\mathbf{x}_{k-1} d\widehat{\mathbf{x}}_{k-1} d\mathbf{x}_k \\ &= \int \ldots \int p(\widehat{\mathbf{x}}_1) \prod_{i=0}^{k-2} p(\widehat{\mathbf{x}}_{k-i}|\widehat{\mathbf{x}}_{k-1-i}, \mathbf{x}_{k-i}) \cdot \\ &\quad \cdot \prod_{i=0}^{k-2} p(\mathbf{x}_{k-i}) d\widehat{\mathbf{x}}_1 \ldots d\widehat{\mathbf{x}}_{k-1} d\mathbf{x}_2 \ldots d\mathbf{x}_k \end{aligned} \tag{5}$$

□

We can observe that the probability of the data $p(\widehat{\mathbf{x}}_k)$, generated by the Teacher after learning k databases depends on the data contained in all previously learnt distributions $\{\mathbf{X}_i | i = 1, \ldots, k\}$, where the past data is reproduced recursively by the Teacher as $\{\widehat{\mathbf{x}}_i | i = 1, \ldots, k-1\}$ after learning sequentially each database.

Definition 1. Let us consider the Wasserstein-1 distance as the Earth-Mover (E-M) distance between a target distribution $p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k)$, and the distribution $p(\widehat{\mathbf{x}}_k)$, generated by the network G_{ψ_k}, as:

$$\begin{aligned} &\mathbf{W}(p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k), p(\widehat{\mathbf{x}}_k)) = \\ &= \sup_{||D||_L \leq 1} \left\{ \mathbb{E}_{\mathbf{x}\sim p(\widehat{\mathbf{x}}_{k-1},\mathbf{x}_k)}[D(\mathbf{x})] - \mathbb{E}_{\mathbf{x}\sim G_{\psi_k}}[D(\mathbf{x})] \right\} \\ &= \sup_{||D||_L \leq 1} \left\{ \mathbb{E}_{\mathbf{x}\sim p(\widehat{\mathbf{x}}_{k-1},\mathbf{x}_k)}[D(\mathbf{x})] - \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z}),\delta\sim p(\delta)}[D(G_{\psi_k}(\mathbf{z},\delta))] \right\}. \end{aligned} \tag{6}$$

Definition 2. We define the following conditional probability of the data generated by the Teacher module implemented by a WGAN network, as :

$$p(\widehat{\mathbf{x}}_k|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) = 1 - \min(1, ||\mathbf{W}(p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k), p(\widehat{\mathbf{x}}_k))). \tag{7}$$

Observation 2. By maximizing the probability density function $p(\widehat{\mathbf{x}}_k|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k)$, defined in equation (7), we maximize the ability of the Teacher module to learn all previously given tasks, including the one defined by the last database $\mathbf{X}_k$.

Proof. We can observe that when fulfilling the objective function during WGAN training, we have

$$\mathbf{W}(p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k), p(\widehat{\mathbf{x}}_k)) = 0, \tag{8}$$

and then

$$p(\widehat{\mathbf{x}}_k|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) = 1, \tag{9}$$

which means that

$$p(\widehat{\mathbf{x}}_k) \approx p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) \tag{10}$$

□

Observation 3. All previously learnt distributions must be the exact approximations of their target distributions in order to allow $p(\widehat{\mathbf{x}}_k)$ to approximate the true joint data distribution $\mathcal{X}$ exactly.

Proof. In order to allow $p(\widehat{\mathbf{x}}_k)$ to approximate the joint distribution, we have:

$$\begin{aligned} &p(\widehat{\mathbf{x}}_k|\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) = 1 \Rightarrow \\ &p(\widehat{\mathbf{x}}_k) = p(\widehat{\mathbf{x}}_{k-1}, \mathbf{x}_k) = p(\widehat{\mathbf{x}}_{k-1}) p(\mathbf{x}_k), \end{aligned} \tag{11}$$

where we consider that $p(\widehat{\mathbf{x}}_{k-1})$ is independent from $p(\mathbf{x}_k)$. Similarly to $p(\widehat{\mathbf{x}}_k)$, we have $p(\widehat{\mathbf{x}}_{k-1}|\widehat{\mathbf{x}}_{k-2}, \mathbf{x}_{k-1}) = 1$, which results in $p(\widehat{\mathbf{x}}_{k-1}) = p(\widehat{\mathbf{x}}_{k-2}) p(\mathbf{x}_{k-1})$. Recursively, following mathematical induction, we have:

$$\prod_{i=0}^{k-2} p(\widehat{\mathbf{x}}_{k-i}|\widehat{\mathbf{x}}_{k-i-1}, \mathbf{x}_{k-i}) = 1 \Rightarrow p(\widehat{\mathbf{x}}_k) = p(\mathbf{x}_1, \ldots, \mathbf{x}_k) \tag{12}$$

□

When considering WGAN for the Teacher module, we fulfil equations (8) and (12). Then, $p(\widehat{\mathbf{x}}_k)$ approximates the true joint distribution $\prod_{i=1}^{k} p(\mathbf{x}_i)$. The scheme of the Teacher module is illustrated in the upper section of Figure 1. The assumptions of Observations 1, 2 and 3 is that we have an ideal generator as Teacher. However, in reality we are using real learning machines bound by physical limitations, and these limitations are discussed in Section 6.

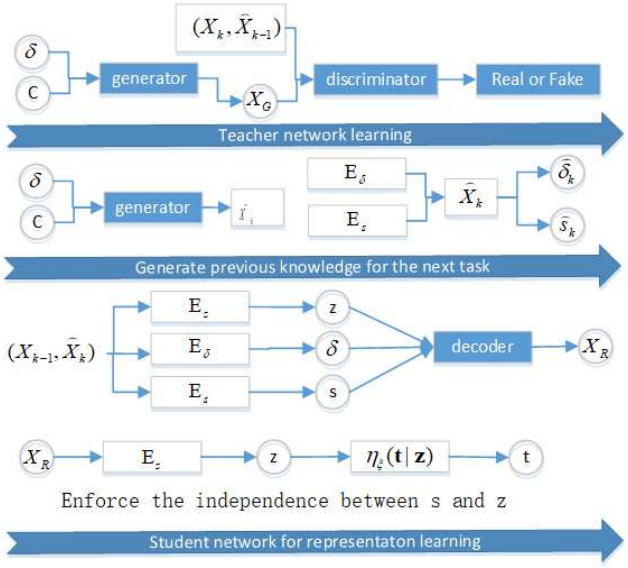


Fig. 1. The scheme of the Teacher-Student network for lifelong learning.

3.3 Student module

The Student module is implemented by a Variational Autoencoder (VAE), which is fed with the data $\widehat{\mathbf{x}}_k$ generated by the WGAN Teacher module, whose framework was described in the previous section, and with the data sampled from the latest given database for training, $\mathbf{x}_{k+1} \sim \mathbf{X}_{k+1}$.

In probabilistic terms, a VAE aims to represent both input data $\{\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1}\}$ and its characteristic latent space $\mathbf{z}_{k+1}$, when considering learning the probabilistic representation of a new domain $\mathbf{X}_{k+1}$, after having previously learnt those for $\{\mathbf{X}_j | j = 1, \ldots, k\}$, as:

$$\begin{aligned} p(\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1}, \mathbf{z}_{k+1}) &= p(\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1}|\mathbf{z}_{k+1})p(\mathbf{z}_{k+1}) \\ &= p(\mathbf{z}_{k+1}|\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1})p(\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1}). \end{aligned} \tag{13}$$

Observation 4. The latent space variables estimated by the Student VAE network, model a probabilistic representation of the information across all databases learnt during the lifelong learning process.

Proof. Let us consider only the derivation of the latent variables from (13) and replace the probability of the data provided by the Teacher module, $p(\widehat{\mathbf{x}}_k)$ with the expression from equation (5):

$$\begin{aligned} p(\mathbf{z}_{k+1}) &= \iint p(\mathbf{z}_{k+1}|\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1})p(\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1})d\widehat{\mathbf{x}}_k d\mathbf{x}_{k+1} \\ &= \iint p(\mathbf{z}_{k+1}|\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1})p(\widehat{\mathbf{x}}_k)p(\mathbf{x}_{k+1})d\widehat{\mathbf{x}}_k d\mathbf{x}_{k+1} \\ &= \int \ldots \int p(\mathbf{z}_{k+1}|\widehat{\mathbf{x}}_k, \mathbf{x}_{k+1})p(\widehat{\mathbf{x}}_1)p(\mathbf{x}_{k+1}) \cdot \\ &\quad \cdot \prod_{i=0}^{k-2} p(\widehat{\mathbf{x}}_{k-i}|\widehat{\mathbf{x}}_{k-1-i}, \mathbf{x}_{k-i}) \cdot \\ &\quad \cdot \prod_{i=0}^{k-2} p(\mathbf{x}_{k-i}) d\widehat{\mathbf{x}}_1 \ldots d\widehat{\mathbf{x}}_{k-1} d\mathbf{x}_2 \ldots d\mathbf{x}_k d\mathbf{x}_{k+1}. \end{aligned} \tag{14}$$

where we have considered mathematical induction through the recursive learning of several tasks.

The expression from (14) describes the statistical relationships between the generative replay mechanisms and the representation learning processes involved. In the following, we show that if $p(\widehat{\mathbf{x}}_k)$ approximates the true joint distribution exactly, then the latent representation is actually learnt from multiple data distributions. Let us consider the results from Observation 1, in the case of the optimal solution when using a Teacher WGAN network, and after replacing the expressions from (12) and (13) into (14), we have :

$$\begin{aligned} p(\mathbf{z}_{k+1}) = \int \ldots \int p(\mathbf{z}_{k+1}|\mathbf{x}_1, \ldots, \mathbf{x}_{k+1}) \cdot \\ \cdot \prod_{i=1}^{k+1} p(\mathbf{x}_i) d\mathbf{x}_1, \ldots, \mathbf{x}_{k+1} \end{aligned} \tag{15}$$

□

This equation demonstrates that the latent space representation of the Student module is learnt from all previously learnt true data distributions, as stated by Observation 4.

For the Student module, we train a variational posterior $p_\theta(\mathbf{z}, \mathbf{s}, \delta|\mathbf{x})$, modelling a diversity of latent spaces defining continuous $\mathbf{z}$, categorical $\mathbf{s}$, and domain δ, variables, respectively. The latent variable model is learnt by maximizing the evidence lower bound (ELBO) depending on the variational posterior, which provides an approximation to the marginal data log-likelihood :

$$\begin{aligned} \log p(\mathbf{x}) &= \iiint \log q(\mathbf{x}, \mathbf{s}, \delta, \mathbf{z}) \, d\mathbf{s} \, d\delta \, d\mathbf{z} \\ &\geq \mathbb{E}_{p(\mathbf{z},\mathbf{s},\delta|\mathbf{x})} \left[\log \left(\frac{q(\mathbf{x}, \mathbf{s}, \delta, \mathbf{z})}{p(\mathbf{z}, \mathbf{s}, \delta|\mathbf{x})} \right) \right] = L_{Stud} \end{aligned} \tag{16}$$

where L_{Stud} represents the objective function for the Student module. We use appropriate inference models in order to approximate the true posteriors and derive the evidence lower bound on the log-likelihood :

$$\begin{aligned} L_{Stud} &= \mathbb{E}_{p_\theta(\mathbf{z},\mathbf{s},\delta|\mathbf{x})} \left[\log \left(\frac{q_\omega(\mathbf{x}|\mathbf{z}, \mathbf{s}, \delta)p(\mathbf{s})p(\delta)p(\mathbf{z})}{p_\theta(\mathbf{z}, \mathbf{s}, \delta|\mathbf{x})} \right) \right] \\ &= \mathbb{E}_{p_\theta(\mathbf{z},\mathbf{s},\delta|\mathbf{x})} \left\{ \log q_\omega(\mathbf{x}|\mathbf{z}, \mathbf{s}, \delta) + \log \left[\frac{p(\mathbf{z})}{p_{\theta_1}(\mathbf{z}|\mathbf{x})} \right] \right. \\ &\quad \left. + \log \left[\frac{p(\delta)}{p_{\theta_2}(\delta|\mathbf{x})} \right] + \left[\frac{p(\mathbf{s})}{p_{\theta_3}(\mathbf{s}|\mathbf{x})} \right] \right\} = \\ &\mathbb{E}_{p_\theta(\mathbf{z},\mathbf{s},\delta|\mathbf{x})} \log \left[q_\omega(\mathbf{x}|\mathbf{z}, \mathbf{s}, \delta) \right] + \mathbb{E}_{p_{\theta_1}(\mathbf{z}|\mathbf{x})} \log \left[\frac{p(\mathbf{z})}{p_{\theta_1}(\mathbf{z}|\mathbf{x})} \right] \\ &\quad + \mathbb{E}_{p_{\theta_2}(\delta|\mathbf{x})} \left[\frac{p(\delta)}{p_{\theta_2}(\delta|\mathbf{x})} \right] + \mathbb{E}_{p_{\theta_3}(\mathbf{s}|\mathbf{x})} \left[\frac{p(\mathbf{s})}{p_{\theta_3}(\mathbf{s}|\mathbf{x})} \right], \end{aligned} \tag{17}$$

where we consider the independence between the probabilities of the latent variables $p(\mathbf{z})$, $p(\delta)$ and $p(\mathbf{s})$ and ω represents the parameters of the decoder $q_\omega(\mathbf{x}|\mathbf{z}, \mathbf{s}, \delta)$, while θ represents the parameters of the encoder. Therefore we consider three separate encoders, $E_\mathbf{z}$, E_δ and $E_\mathbf{s}$, as illustrated in the scheme fro Figure 1, used for modeling the variational distributions $p_{\theta_1}(\mathbf{z}|\mathbf{x})$, $p_{\theta_2}(\delta|\mathbf{x})$ and $p_{\theta_3}(\mathbf{s}|\mathbf{x})$, defined by the independent parameters θ_1, θ_2, θ_3. Then we can rewrite the objective function for the Student module as that corresponding to a VAE, expressed with respect to

the Kullback-Leibler divergences of the continuous, discrete and domain latent variables, respectively, as:

$$\begin{aligned} L_{Stud} &= \mathbb{E}_{p_\theta(\mathbf{z},\mathbf{s},\delta|\mathbf{x})} \log(q_\omega(\mathbf{x}|\mathbf{z},\mathbf{s},\delta)) \\ &- \beta_1 D_{KL}(p_{\theta_1}(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) \\ &- \beta_2 D_{KL}(p_{\theta_2}(\delta|\mathbf{x})||p(\delta)) - \beta_3 D_{KL}(p_{\theta_3}(\mathbf{s}|\mathbf{x})||p(\mathbf{s})), \end{aligned} \tag{18}$$

where the first term represents the reconstruction error of the data and the following three components represent the KL divergence terms for the continuous latent variables $\mathbf{z}$, discrete latent space δ, and the variables corresponding to the continuous latent space $\mathbf{s}$, and β_1, β_2 and β_3 represent their contributions to L_{Stud}. The distribution of the continuous variables is modelled as Gaussian, $p_{\theta_1}(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mu, \sigma)$. We use the reparameterization trick [30], [31], in order to generate differentiable samples from $p_{\theta_1}(\mathbf{z}|\mathbf{x})$, as

$$\mathbf{z} = \mu(\mathbf{x}) + \sigma(\mathbf{x}) \odot \mathcal{N}(0, \mathbf{I}). \tag{19}$$

The probabilities $p(\mathbf{s})$ and $p(\delta)$ and are the priors of the discrete $\mathbf{s}$ and categorical δ represent latent variables. Parameterizing $p_{\theta_3}(\mathbf{s}|\mathbf{x})$ and $p_{\theta_2}(\delta|\mathbf{x})$ is challenging given that categorical distributions are non-differentiable and cannot be updated when integrated into a network trained using Stochastic Gradient Descent (SGD). Consequently, the two conditional distributions $p_{\theta_2}(\delta|\mathbf{x})$ and $p_{\theta_3}(\mathbf{s}|\mathbf{x})$, from (18) are approximated using two distinct encoders, each modelled by a Gumbel-softmax distribution [39], [40], representing a categorical distribution which is differentiable and can be used for inferring random categorical variables:

$$s_j = \frac{\exp((\log a_j + g_j)/T)}{\sum_{i=1}^{L_m} \exp((\log a_i + g_i)/T)} \tag{20}$$

for $j = 1, \ldots, L_m$, where $\{a_1, a_2, \ldots, a_{L_m}\}$ represent the discrete variable (for example class labels) probabilities for L_m classes of m-th database. g_j is sampled from the $\text{Gumbel}(0, 1)$ distribution, and T is a temperature parameter which controls the degree of relaxation.

One issue in VAEs is whether to consider a fixed prior distribution $p(\mathbf{z})$ for the latent space field $\mathbf{z}$ or a conditional distribution on certain factors. Data from different domains (defining various tasks) may contain both shared and specific generative factors. Data from the same class will share specific characteristics. In the following we consider using a conditional prior distribution for the continuous latent variable $\mathbf{z}$ on the domain variable δ in order to introduce domain-specific generative factors :

$$p(\mathbf{z}|\delta) = \mathcal{N}(f(\delta), \sigma^2\mathbf{I}), \tag{21}$$

where $f(\delta)$ is a transforming function which uses a one-hot vector to select a single discrete variable. The domain variables δ are estimated from the past learnt data sets in which domain labels are known. By using such priors we can group the data according to their task information.

4 Training the Teacher-Student network for lifelong learning

The Lifelong Teacher-Student (LTS) structure is illustrated in the diagram from Figure 1. The Student module is implemented by three encoders, each assigned for modelling a specific type of latent variable: $E_{\mathbf{z}}$ for the continuous variables, E_δ for the domain variables, and $E_{\mathbf{s}}$ for the discrete variables, respectively, and a single decoder network, as illustrated in Figure 1. The encoders are characterized by the parameters θ_1, θ_2 and θ_3, while the decoder is characterised by the ω parameters. The Student module is trained by maximizing its characteristic ELBO objective function L_{Stud}, from (18). In order to encourage the encoders to learn discrete meaningful representations of data, we introduce two cross-entropy loss functions for the discrete-specific encoder and domain-specific encoder, respectively:

$$L_{\mathbf{s}} = \mathbb{E}_{(\mathbf{x},\delta,\mathbf{y})\sim(\mathcal{X},\mathcal{D},\mathcal{Y})} \eta(p_{\theta_3}(\mathbf{s}|\mathbf{x}), \mathbf{y}) \tag{22}$$

$$L_\delta = \mathbb{E}_{(\mathbf{x},\delta,\mathbf{y})\sim(\mathcal{X},\mathcal{D},\mathcal{Y})} \eta(p_{\theta_2}(\delta|\mathbf{x}), \delta) \tag{23}$$

where $\eta(\cdot)$ is the cross-entropy loss depending on $\mathbf{s}$ or δ, which represents the categorical domain such as class labels, and the domain labels, depending on the specific task being learnt whose probabilistic spaces are denoted as $\mathcal{Y}$ and $\mathcal{D}$, respectively, while $\mathbf{x}$ are the training data. The training data for the Student module includes the data sampled from the latest available data set $\mathbf{x}_k \sim \mathbf{X}_k$, as well as the data generated by the Teacher module $\widehat{\mathbf{x}}_{k-1}$, corresponding to the previously taught probabilistic representations of databases $\{\mathbf{X}_i | i = 1, \ldots, k-1\}$.

The data from the current database $\mathbf{x}_k$, to be used for training, and the generated data $\widehat{\mathbf{x}}_{k-1}$, are incorporated into a single data set, which is used for training the Student module, as explained in Section 3.3, at a ratio depending on the importance of the old tasks relative to those corresponding to a new task. In the experimental results we consider equal ratios for sampling data from a new task $\mathbf{x}_k \sim \mathbf{X}_k$ with the generated data $\widehat{\mathbf{x}}_{k-1}$, corresponding to older tasks. Then, the VAE network is trained to represent and reconstruct the whole cumulative learning space $p(\mathbf{z}_k)$, which represents the information from all previously given databases, as supported by Observation 4. The two specific encoders are trained using the cross-entropy loss functions $L_{\mathbf{s}}$ and L_δ from (22) and (23), respectively.

The Teacher WGAN network is used to replay past data samples associated with the previous tasks, and its training data are identical to those used for training the Student module, starting with the second database, $\mathbf{X}_2$. During the training of the Teacher module we also encourage the independence between the variables $\mathbf{z}$ and $\mathbf{s}$ representing the continuous variation of data and discrete specific information, respectively. In order to achieve this, we introduce a new variable $\mathbf{t}$ which has the same dimension with $\mathbf{s}$, predicted by an auxiliary encoder, defined by $\eta_\zeta(\mathbf{t}|\mathbf{z})$, from the continuous latent representation $\mathbf{z}$, as shown in the lower section of the model's diagram from Figure 1. The loss function used for training $\eta_\zeta(\mathbf{t}|\mathbf{z})$ and $p_{\theta_1}(\mathbf{z}|\mathbf{x})$ is defined as the cross-entropy loss L, optimized as, [41], [42], [43]:

$$\max_{\theta_1} \min_{\zeta} L(\eta_\zeta(\mathbf{t}|p_{\theta_1}(\mathbf{z}|\mathbf{x})), \mathbf{s}) \tag{24}$$

where $p_{\theta_1}(\mathbf{z}|\mathbf{x})$, is a component of L_{Stud} from (18), defined by the encoders modelling the continuous latent variables. Optimizing this loss function enforces the independence between $\mathbf{s}$ and $\mathbf{z}$ encouraging the continuous latent representations to capture specific non-class information from the data.

The pseudocode of the training algorithm for the lifelong Teacher-Student network is provided in Algorithm 1.

Algorithm 1: The training algorithm for the Teacher-Student framework.

Initial training phase:

1:Sample $X^1 = \{x_1^1, x_2^1, ..., x_N^1\}$ from the first task

2:Sample $Y^T = \{y_1^1, y_2^1, ..., y_N^1\}$ from the first task

3:Assign $\delta^T = \{\delta_1^1, \delta_2^1, ..., \delta_N^1\}$ for the first task

4:**While** *epoch* < *epoch*$^{\max}$ **do**

5: **While** *batch* < *batch*$^{\max}$ **do** minibatch procedure

6: $\boldsymbol{x}_{batch} = Select\left(epoch, X^1\right)$ batch samples

7: $y_{batch} = Select\left(epoch, Y^1\right)$ batch samples

8: Train the teacher network by using adversarial loss

9: Train the student network by optimizing L_{Stud}

10: Train the class-specific and domain-specific encoders by L_s, L_δ

11: Separate z and s by $\max_{\theta_1} \min_{\varsigma} L\left(\eta_\varsigma\left(s \mid p_{\theta_1}(z \mid x)\right), s\right)$

12: **End**

13:**End**

Following training phase:

14:Sample $X^T = \{x_1^T, x_2^T, ..., x_N^T\}$ from the T-th task

15:Sample $Y^T = \{y_1^T, y_2^T, ..., y_N^T\}$ from the T-th task

16:Assign $\delta^T = \{\delta_1^1, \delta_2^1, ..., \delta_N^1\}$ for the T-th task

14:Sample $\{X^1, .., X^{T-1}\} = \{x_1^1, x_2^2, ..., x_N^{T-1}\}$ from the teacher

15:Obtain $\{Y^1, .., Y^{T-1}\} = \{y_1^1, y_2^2, ..., y_N^{T-1}\}$ inferred by the student

16:Obtain $\{\delta^1, ..\delta^{T-1}\} = \{\delta_1^1, \delta_2^2, ..., \delta_N^{T-1}\}$ inferred by the student

18: $\{X^{mix}, Y^{mix}, D^{mix}\} = \{X^{past}, Y^{past}, D^{past}\} \cup \{X^2, Y^2, D^2\}$

19: $X_{\text{Joint}} = X^T \cup \{X^1, .., X^{T-1}\}$

20: $Y_{\text{Joint}} = Y^T \cup \{Y^1, .., Y^{T-1}\}$

21: $\delta_{\text{Joint}} = \delta^T \cup \{\delta^1, .., \delta^{T-1}\}$

22:**While** *epoch* < *epoch*$^{\max}$ **do**

23: **While** *batch* < *batch*$^{\max}$ **do** minibatch procedure

24: Train the teacher network by using adversarial loss

25: Train the student network by optimizing L_{Stud}

26: Train the class-specific and domain-specific encoders by L_s, L_δ

27: Separate z and s by $\max_{\theta_1} \min_{\varsigma} L\left(\eta_\varsigma\left(t \mid p_{\theta_1}(z \mid x)\right), s\right)$

27: **End**

28:**End**

5 Semi-supervised and unsupervised lifelong learning

The proposed approach can also be extended to be applied under the semi-supervised learning framework. Kingma *et al.* [44] introduced a VAE framework for semi-supervised learning in which the model uses both labeled and unlabeled data samples during training. In this paper, we extend the proposed approach to deal with semi-supervised problems under the lifelong learning setting. We consider that only a small part of the current training set is labelled, while the labels for the other data would be inferred by the model, following training.

In the following, two distinct situations are considered. In the first case, the class label $\mathbf{s}$ is available and we simply incorporate the class information during the decoding stage without involving any inference. Then, the variational bound for the VAE is defined as :

$$L_{SVAE} = \mathbb{E}_{p_{\theta_1,\theta_2}(\mathbf{z},\delta|\mathbf{x}),\mathbf{s}\sim p(\mathbf{s})}(q_\omega(\mathbf{x}|\mathbf{z},\mathbf{s},\delta)) - \\ - \beta_1 D_{KL}(p_{\theta_1}(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) - \beta_2 D_{KL}(p_{\theta_2}(\delta|\mathbf{x})||p(\delta)) \tag{25}$$

where we only infer continuous and domain latent variables $\mathbf{z}$ and δ, by considering $p_{\theta_1}(\cdot)$ and $p_{\theta_2}(\cdot)$, while the variable $\mathbf{s}$ is associated with the class label. The latent variables $\mathbf{s}$ are marginally independent, encouraging the separation of the class specification from other continuous variations. In the second case we consider that the class label y is missing, aiming for this to be inferred by the class-specific encoder. The variational bound for unobserved data is defined as :

$$L_{UVAE} = \mathbb{E}_{p_{\theta_1}(\mathbf{z},\mathbf{s},\delta|\mathbf{x})}(q_\omega(\mathbf{x}|\mathbf{z},\mathbf{s},\delta)) \\ - \beta_1 D_{KL}(p_{\theta_1}(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) - \beta_2 D_{KL}(p_{\theta_2}(\delta|\mathbf{x})||p(\delta)) \\ - \beta_3 D_{KL}(p_{\theta_3}(\mathbf{s}|\mathbf{x})||p(\mathbf{s})). \tag{26}$$

For the semi-supervised learning we consider the cross-entropy loss, for a set of labeled data, as in equation (22), as well as for the domain data, as in (23). Then, the full loss for semi-supervised learning is defined by combining L_{SVAE} and L_{UVAE} from (25) and (26):

$$L_{SemiSupVAE} = L_{SVAE} + aL_{UVAE} \tag{27}$$

where a controls the importance of the unsupervised versus the supervised component of the loss.

In addition to the semi-supervised and supervised learning tasks, this paper also extends the proposed approach for the unsupervised learning setting. In this case, there are no class labels for any of the given data, and we only consider two encoders $p_{\theta_1}(\mathbf{z}|\mathbf{x})$ and $p_{\theta_2}(\delta|\mathbf{x})$ in the objective function L_{Stud} from (18). We train the Student module to approximate the joint data distribution, by maximizing the ELBO, as :

$$L_{VAE2} = \mathbb{E}_{p_{\theta_1}(\mathbf{z},\delta|\mathbf{x})}(q_\omega(\mathbf{x}|\mathbf{z},\delta)) - \beta_1 D_{KL}(p_{\theta_1}(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) \\ - \beta_2 D_{KL}(p_{\theta_2}(\delta|\mathbf{x})||p(\delta)). \tag{28}$$

The β-VAE model [4] was shown to be successful for the unsupervised visual disentangled representations learning. This model modifies the VAE objective function by imposing a large penalty β on the KL term [45], thus encouraging disentanglement in the latent variable space. β-VAE is also adopted in this study in order to enable data disentanglement under the lifelong Teacher-Student learning. We set $\beta_1 = 1$ and $\beta_2 = 1$ when generating images and increase the value of β_1 for achieving disentangled representations. We consider the prior $p(\mathbf{z})$ to be conditioned on the domain variable δ, according to equation (21), which encourages the posteriors $p_{\theta_1}(\mathbf{z}|\mathbf{x})$ defined by the inference model, given the data $\mathbf{x}$ with the associated domain variable δ, inferred by $p_{\theta_2}(\delta|\mathbf{x})$, to be projected into several distinct clusters in the latent space. This property determines the Student module to capture different underlying factors, in its latent space representation, for each domain.

6 The error bounds for the lifelong learning of the Student module

An essential aspect of lifelong learning systems is their ability to learn new tasks, corresponding to diverse sets

of data, without forgetting. In this Section we provide a theoretical analysis into how the proposed VAE Student model can remember or conversely, forget, previously learnt knowledge during the lifelong learning process and the limitations of the proposed model. The theoretical analysis is inspired by the domain adaptation theory [10], [46], where error bounds are evaluated for the transfer of information from one data domain to another in learning systems.

Let us consider the association $(\mathcal{X}, \mathcal{Y})$ between the input data space $\mathcal{X}$ and the outputs $\mathcal{Y}$. Let $\mathcal{D}^k = \{\mathbf{x}_i^k, y_i^k | i = 1, \ldots, N_k\}$ be a data distribution drawn from the k-th testing set, when learning its corresponding task. Similarly, $\widetilde{D}^k$ represents a training distribution set from the k-th task. Let $\widehat{\mathcal{D}}^k = \{\widehat{\mathbf{x}}_i^k, \widehat{y}_i^k | i = 1, \ldots, \widehat{N}_k\}$ represent the joint distribution $\widehat{\mathcal{D}}^k = \widetilde{\mathcal{D}}^k \cup p(\widehat{\mathbf{x}}_{k-1})$, between the data generated from $p(\widehat{\mathbf{x}}_{k-1})$ by the Teacher, after previously learning the probabilistic representations of all other training sets including $\widehat{\mathcal{D}}^{k-1}$, and the data corresponding to the training set, sampled from the new task, $\widetilde{\mathcal{D}}^k$. Each data sample $\widehat{\mathbf{x}}_i^{k-1} \sim p(\widehat{\mathbf{x}}_{k-1})$ is generated by the Teacher model and each label $\widehat{y}_i^{k-1}$ is predicted by the Student model. Let $h(\cdot)$ represent a mapping $h : \mathcal{X} \rightarrow \mathcal{Y}$, which corresponds to $p_{\theta_3}(\mathbf{s}|\mathbf{x})$, one of the components in the Student's objective function L_{Stud} from equation (18).

In the following we define a loss function, $\psi : \mathcal{Y} \times \mathcal{Y} \rightarrow [0, 1]$ that gives a cost of $h(\mathbf{x})$, deviating from the true output $y \in \mathcal{Y}$, [47].

Definition 3 (Empirical risk). For a given loss function $\psi : \mathcal{Y} \times \mathcal{Y} \rightarrow [0, 1]$ and a training set $\{\mathbf{x}_i, y_i \sim \mathcal{D} | i = 1, \ldots, m\}$, the empirical risk for a given hypothesis $h \in \mathcal{H}$ is defined as:

$$\mathrm{R}_{\mathcal{D}}(h) = \frac{1}{m} \sum_{i=1}^{m} \psi(h(\mathbf{x}_i), y_i), \tag{29}$$

and for a pair of hypotheses $(h, h') \in \mathcal{H}^2$, we consider the notation $\mathrm{R}_{\mathcal{D}}(h, h') = \sum_{i=1}^{m} \psi(h(\mathbf{x}_i), h'(\mathbf{x}_i))/m$.

Definition 4 (Discrepancy distance). Given two domains $\mathcal{D}$ and $\widehat{\mathcal{D}}$ over $\mathcal{X} \times \mathcal{Y}$, let $\psi : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathcal{R}_+$ represent a loss function. Let $\mathcal{D}_{\mathcal{X}}$ and $\widehat{\mathcal{D}}_{\mathcal{X}}$ represent marginals on $\mathcal{D}$ and $\widehat{\mathcal{D}}$. The discrepancy distance Δ between two marginals is defined as:

$$\begin{aligned} \Delta_{\psi}(\mathcal{D}_{\mathcal{X}}, \widehat{\mathcal{D}}_{\mathcal{X}}) = \sup_{h, h'} \big| & \mathbb{E}_{\mathbf{x} \sim \mathcal{D}_{\mathcal{X}}} \left[\psi(h'(\mathbf{x}), h(\mathbf{x}))\right] \\ & - \mathbb{E}_{\mathbf{x} \sim \widehat{\mathcal{D}}_{\mathcal{X}}} \left[\psi(h'(\mathbf{x}), h(\mathbf{x}))\right] \big| \end{aligned} \tag{30}$$

where $h(\cdot)$ and $h'(\cdot)$ are mappings defined on the domains $\mathcal{D}$ and $\widehat{\mathcal{D}}$.

Theorem 1. Let $\mathcal{D}_{\mathcal{X}}$ and $\widehat{\mathcal{D}}_{\mathcal{X}}$ represent marginals on $\mathcal{D}$ and $\widehat{\mathcal{D}}$, while $f \in \mathcal{H} : \mathcal{X} \rightarrow \mathcal{Y}$ represents the true labeling function. The relationship between these marginals is defined by:

$$\mathrm{R}_{\mathcal{D}_{\mathcal{X}}}(h, f) \leq \mathrm{R}_{\widehat{\mathcal{D}}_{\mathcal{X}}}(h, f_{\widehat{\mathcal{D}}_{\mathcal{X}}}) + \Delta_{\psi}(\mathcal{D}_{\mathcal{X}}, \widehat{\mathcal{D}}_{\mathcal{X}}) + \lambda(\mathcal{D}_{\mathcal{X}}, \widehat{\mathcal{D}}_{\mathcal{X}}), \tag{31}$$

where $\lambda(\mathcal{D}_{\mathcal{X}}, \widehat{\mathcal{D}}_{\mathcal{X}})$ is the combined error term defined as :

$$\lambda(\mathcal{D}_{\mathcal{X}}, \widehat{\mathcal{D}}_{\mathcal{X}}) = \mathrm{R}_{\mathcal{D}_{\mathcal{X}}}(h, f_{\mathcal{D}_{\mathcal{X}}}) + \mathrm{R}_{\widehat{\mathcal{D}}_{\mathcal{X}}}(f_{\mathcal{D}_{\mathcal{X}}}, f_{\widehat{\mathcal{D}}_{\mathcal{X}}}), \tag{32}$$

where $f_{\mathcal{D}_{\mathcal{X}}}, f_{\widehat{\mathcal{D}}_{\mathcal{X}}} \in \mathcal{H}$ are two optimal hypotheses, defined as

$$f_{\mathcal{D}_X} = \arg\min_{h \in \mathcal{H}} \mathrm{R}_{\mathcal{D}}(h) \; ; \; f_{\widehat{\mathcal{D}}_X} = \arg\min_{h \in \mathcal{H}} \mathrm{R}_{\widehat{\mathcal{D}}}(h), \tag{33}$$

and

$$\mathrm{R}_{\widehat{\mathcal{D}}_{\mathcal{X}}}(f_{\mathcal{D}_{\mathcal{X}}}, f_{\widehat{\mathcal{D}}_{\mathcal{X}}}) = \mathbb{E}_{\mathbf{x} \sim \widehat{\mathcal{D}}_{\mathcal{X}}} \left[\psi(f_{\mathcal{D}_{\mathcal{X}}}(\mathbf{x}), f_{\widehat{\mathcal{D}}_{\mathcal{X}}}(\mathbf{x}))\right]. \tag{34}$$

The detailed proof is provided in [48]. From this theorem, we find that the risk for $h(\cdot)$ on the target distribution is bounded by the risk for $h(\cdot)$ on the source distribution generated by the Teacher plus the discrepancy distance between the empirical distribution and the generator distribution, provided in Definition 4. In order to analyse how the Teacher forgets previously learnt knowledge during lifelong learning process, we derive an analytical bound in the following theorem.

Theorem 2. From Theorem 1, we can estimate the accumulated errors across K tasks, by deriving an upper bound:

$$\begin{aligned} \sum_{i=1}^{K} \mathrm{R}_{\mathcal{D}_{\mathcal{X}}^{1:i}}(h, f) \leq \sum_{i=1}^{K} \Big(& \mathrm{R}_{\widehat{\mathcal{D}}_{\mathcal{X}}^{i}}(h, f_{\widehat{\mathcal{D}}_{\mathcal{X}}^{i}}) + \Delta_{\psi}(\mathcal{D}_{\mathcal{X}}^{1:i}, \widehat{\mathcal{D}}_{\mathcal{X}}^{i}) \\ & + \lambda(\mathcal{D}_{\mathcal{X}}^{1:i}, \widehat{\mathcal{D}}_{\mathcal{X}}^{i}) \Big), \end{aligned} \tag{35}$$

where $\mathcal{D}_{\mathcal{X}}^{1:i}$ represents the joint distribution of all given databases, $\mathcal{D}_{\mathcal{X}}^{1:i} = \{\mathcal{D}_{\mathcal{X}}^{1} \cup \mathcal{D}_{\mathcal{X}}^{2} \cup \cdots \cup \mathcal{D}_{\mathcal{X}}^{i}\}$ and $\widehat{\mathcal{D}}^1$ represents $\mathcal{D}^1$ for the sake of simplicity.

Proof : Firstly, we consider the learning of the first task and we have a bound for $\mathrm{R}_{\mathcal{D}_{\mathcal{X}}^{1}}(h, f)$, in (31), according to Theorem 1. Similarly, we derive the bound when learning the next task :

$$\begin{aligned} \mathrm{R}_{\mathcal{D}_{\mathcal{X}}^{1:2}}(h, f) \leq & \mathrm{R}_{\widehat{\mathcal{D}}_{\mathcal{X}}^{2}}(h, f_{\widehat{\mathcal{D}}_{\mathcal{X}}^{2}}) + \Delta_{\psi}(\mathcal{D}_{\mathcal{X}}^{1:2}, \widehat{D}_{\mathcal{X}}^{2}) \\ & + \lambda(\mathcal{D}_{\mathcal{X}}^{1:2}, \widehat{\mathcal{D}}_{\mathcal{X}}^{2}). \end{aligned} \tag{36}$$

By mathematical induction, we have the risk corresponding to $\mathcal{D}_{\mathcal{X}}^{1:i}$, after learning the i-th task :

$$\begin{aligned} \mathrm{R}_{\mathcal{D}_{\mathcal{X}}^{1:i}}(h, f) \leq & \mathrm{R}_{\widehat{\mathcal{D}}_{\mathcal{X}}^{i}}(h, f_{\widehat{\mathcal{D}}_{\mathcal{X}}^{i}}) + \Delta_{\psi}(\mathcal{D}_{\mathcal{X}}^{1:i}, \widehat{\mathcal{D}}_{\mathcal{X}}^{i}) \\ & + \lambda(\mathcal{D}_{\mathcal{X}}^{1:i}, \widehat{\mathcal{D}}_{\mathcal{X}}^{i}) \end{aligned} \tag{37}$$

where $i = 1, \ldots, K$. We then sum up all inequalities, resulting in equation (35), which proves Theorem 2 □

From Theorem 2, we find that the minimization of the discrepancy distance (30) between the generator distribution and the target distribution, when learning each task, from a set of different tasks, plays an important role for reducing the risks for $h(\cdot)$ on the true target distribution. This bound can be tight when the Teacher approximates the joint distribution $p(\widehat{\mathbf{x}}_{i-1}) \cup \widetilde{\mathcal{D}}^i$ after each i-th task learning. In this case, the Teacher can generate a true joint distribution $\{\widetilde{\mathcal{D}}^1, \ldots, \widetilde{\mathcal{D}}^K\}$ (see Observation 3). The lifelong learning is then transformed into a multiple source-target domain adaptation problem where the Student is trained on $\{\widetilde{\mathcal{D}}^1, \ldots, \widetilde{\mathcal{D}}^K\}$ and is evaluated on $\{\mathcal{D}^1, \ldots, \mathcal{D}^K\}$. In contrast, if the Teacher can not approximate the joint distribution well, the performance of the Student module for the target distribution depends on the quality of the generative ability of the Teacher module. Additionally, we use WGAN for our Teacher module instead of VAEs due to several reasons. VAE [30] adopts a simple and fixed prior which can not represent exactly the true posterior $p(\mathbf{z}|\mathbf{x})$ [49], leading to vague generation results [50]. In contrast, WGAN [38] aims to minimize the Wasserstein distance between the target and the generator distribution (3), which enjoys the theoretical guarantee on the convergence. The learning

process of WGAN is more stable than that for classical GANs [38], which is why we use it for our Teacher module, requiring to approximate jointly, the previously learnt distributions as well as the distribution corresponding to learning a new task.

7 Experimental results

In the following, we apply the proposed Lifelong Teacher-Student (LTS) learning framework on various tasks. For the hyperparameter setting, we consider $\beta_1 = 1$, while β_2, β_3 are set to very small values in the Student module objective function L_{Stud} from equation (18). According to this objective function, the experiments do not only focus on learning classification tasks but they also aim to learn disentangled data representations, under the lifelong learning setting. We consider three distinct lifelong classification learning experiments: successive learning of similar data distributions, learning of completely different data distributions and semi-supervised lifelong learning. The results for each of these applications are presented in the Subsections 7.1, 7.2 and 7.3, respectively. We also evaluate the representation ability of the proposed approach in unsupervised lifelong learning, where we consider both similar and distinct domains.

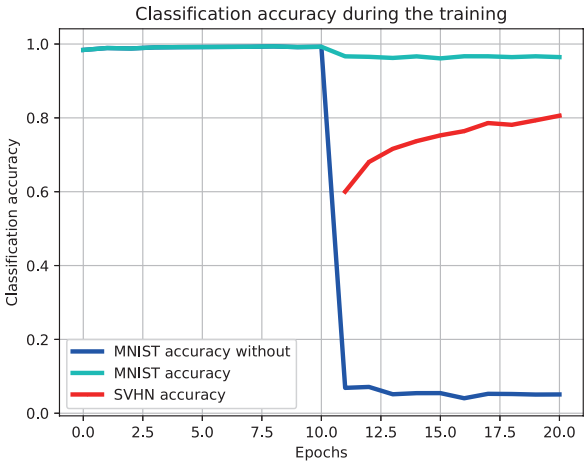
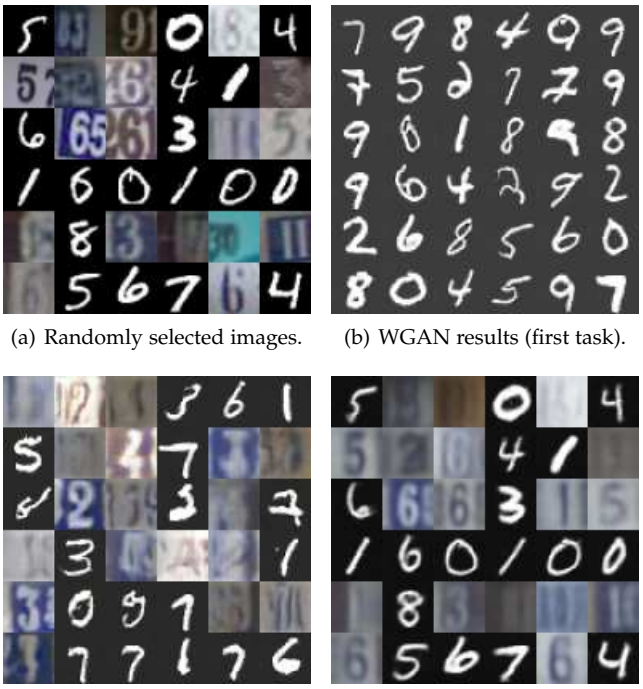


Fig. 2. Classification accuracy curves during the lifelong Teacher-Student learning from MNIST to SVHN databases.

7.1 The lifelong learning of similar domains

In this experiment, we consider the lifelong learning when the proposed LTS framework is aiming to learn two similar domains. We consider MNIST [51] and SVHN [52] databases, both containing images of digits. MNIST data set consists of 60,000 training and 10,000 testing samples, while SVHN consists of 73,257 training and 26,032 testing digital images. We resize the MNIST images into $32 \times 32 \times 3$ pixels resolution. We use a simple CNN consisting of two convolution layers for both the decoder and encoder of the Student module and train it for 10 epochs for MNIST and SVHN, respectively, under the lifelong LTS learning, considering a learning rate of 0.001. The classification accuracy achieved during each epoch is shown in Fig. 2. We can observe from this plot that the performance of the proposed approach



(a) Randomly selected images. (b) WGAN results (first task).

(c) WGAN results from MNIST and SVHN databases. (d) VAE reconstructions from MNIST and SVHN databases.

Fig. 3. Image generation and reconstruction results for the LTS model when learning MNIST and then SVHN databases.

on MNIST would not decrease much when learning an additional task such as SVHN. However, when not using the reply of the first database by the GAN Teacher network (marked as "MNIST accuracy without"), the performance drops significantly. A set of images, selected randomly from MNIST and SVHN datases are shown in Fig. 3a. Images generated by the WGAN Teacher network, after learning the information corresponding to a single database MNIST, are shown in Fig. 3b, while the reconstructed images by WGAN Teacher and VAE Student networks, considering the lifelong learning of MNIST and SVHN distributions, are provided in Figs. 3c and 3d, respectively. For comparison we consider the Lifelong Generative Modeling (LGM) [33], with the same network architecture as for the LTS approach. We also consider MemoryGAN [53] for comparison and the numerical results are provided in Table 1, where S-M indicates the lifelong learning when considering the databases in reversed order, firstly SVHN and then MNIST. The results from Table 1 indicate that LTS achieves the best results in most cases.

7.2 The lifelong learning of different domains

We evaluate the performance of the proposed LTS model on two completely different domains. After MNIST we consider MNIST-Fashion [54] dataset, with the same number of training and testing images as for the former database. MNIST-Fashion contains 10 classes of images representing different clothing items, of shape and characteristics which are completely different from those of the images from MNIST. We adopt the same network architecture and hyperparameter setting for both the proposed LTS, and the lifelong learning approach LGM, [33]. The classification curves,

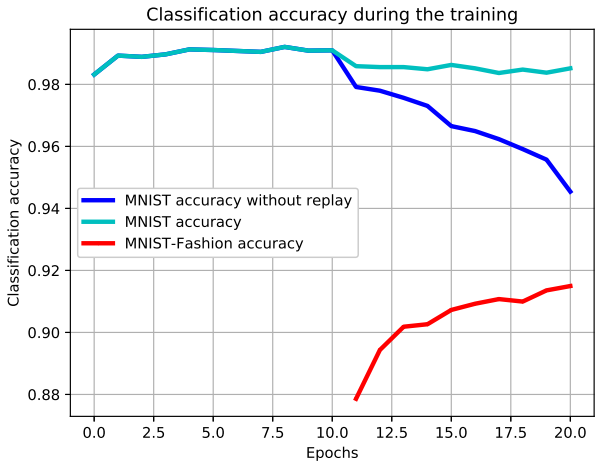


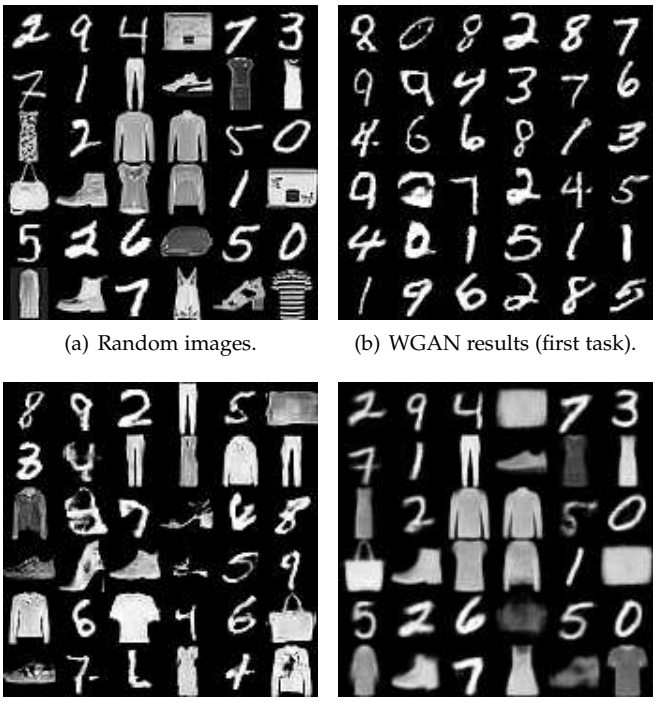
Fig. 4. Classification accuracy curves during the lifelong Teacher-Student learning from MNIST to MNIST-Fashion databases.

TABLE 1
Classification accuracy when learning MNIST and SVHN under the lifelong learning setting.

Methods	Testing data set	Lifelong	Accuracy
LTS	MNIST	M-S	96.66
MemoryGANs [53]	MNIST	M-S	96.04
LGM [33]	MNIST	M-S	96.59
LTS	SVHN	M-S	80.15
MemoryGANs [53]	SVHN	M-S	80.03
LGM [33]	SVHN	M-S	80.77
LTS	MNIST	S-M	98.80
MemoryGANs [53]	MNIST	S-M	98.29
LGM [33]	MNIST	S-M	98.56
LTS	SVHN	S-M	80.39
MemoryGANs [53]	SVHN	S-M	79.34
LGM [33]	SVHN	S-M	76.76

for the lifelong learning of MNIST to MNIST-Fashion are shown in Fig. 4, considering 10 epochs for training the models successively with each database. We also provide the performance of the proposed approach without data replay. From these results we observe that the performance of LTS on MNIST drops slightly when learning MNIST-Fashion as a second database. However, when not considering data replay there is a significant drop on the performance for the former task. A selection of random images from both MNIST and Fashion databases are shown in Fig. 5a, the generated results by WGAN for the first dataset MNIST are provided in Fig. 5b, while the images generated by WGAN Teacher and by the Student VAE, after learning the second database MNIST-Fashion, are shown in Figs. 5c and 5d, respectively. The quality of the images reconstructed by both Student VAE and Teacher WGAN is good despite the radical differences between the images of the two databases.

The classification accuracy of the proposed LTS approach is provided in Table 2, where M-F denotes MNIST to MNIST-Fashion database lifelong learning, while F-M indicates their learning in reversed order. It can be observed that the proposed approach achieves higher classification accuracy than LGM [33] and MemoryGANs [53], on MNIST and MNIST-Fashion under both M-F and F-M settings.



(a) Random images. (b) WGAN results (first task). (c) WGAN results after training on MNIST and MNIST-Fashion. (d) VAE reconstructions from MNIST and MNIST-Fashion.

Fig. 5. The generation and reconstruction results for LTS considering the lifelong learning from MNIST to MNIST-Fashion.

TABLE 2
Classification accuracy on the MNIST and MNIST-Fashion under the lifelong learning setting.

Methods	Testing data set	Lifelong	Accuracy
LTS	MNIST	M-F	98.51
LGM [33]	MNIST	M-F	97.29
MemoryGANs [53]	MNIST	M-F	98.15
LTS	MNIST-Fashion	M-F	91.49
LGM [33]	MNIST-Fashion	M-F	91.71
MemoryGANs [53]	MNIST-Fashion	M-F	91.35
LTS	MNIST	F-M	98.42
LGM [33]	MNIST	F-M	98.85
MemoryGANs [53]	MNIST	F-M	98.52
LTS	MNIST-Fashion	F-M	89.35
LGM [33]	MNIST-Fashion	F-M	86.05
MemoryGANs [53]	MNIST-Fashion	F-M	89.13

7.3 Semi-supervised lifelong learning

We also apply the proposed framework for semi-supervised lifelong learning, where the training is defined by the cost function $L_{SemiSupVAE}$ from equations (27), (25), (26), and (22), where $a = 1.0$. We divide MNIST dataset into two subsets representing labelled and unlabelled images, considering fewer images in the labelled set than for the unlabelled set. For the labelled images we consider an identical number of images for each class. The proposed LTS model is trained firstly on MNIST by considering that only a small number of labelled images is available during the initial learning stage. After training on MNIST, we consider that all generated data are assigned with class labels, inferred by the model and then we train with the second database, MNIST-Fashion.

The semi-supervised classification learning curves for

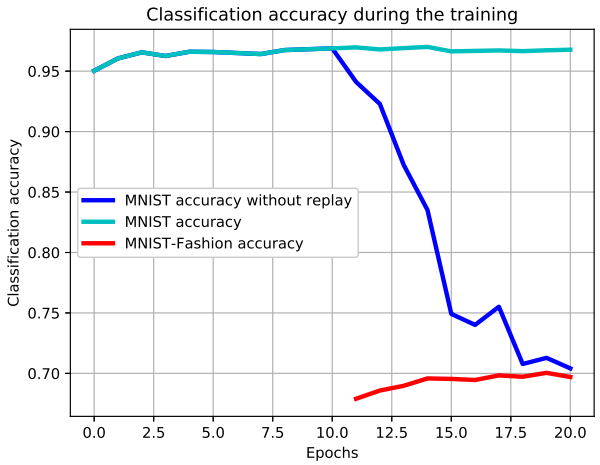


Fig. 6. Semi-supervised classification results from MNIST to MNIST-Fashion. We use 1,000 images from MNIST database and another 10,000 from MNIST-Fashion as a labeled data set.

TABLE 3
Semi-supervised classification results on MNIST data, when considering MNIST to MNIST-Fashion lifelong learning.

Methods	Lifelong?	Error
LTS	Yes	3.18
LGAN [34]	Yes	4.87
Neural networks (NN) [55]	No	10.7
Convolution networks (CNN) [55]	No	6.45
TSVM [55]	No	5.38
CAE [55]	No	4.77
M1+TSVM [55]	No	4.24
M2 [55]	No	3.60
M1+M2 [55]	No	2.40
Semi-VAE [44]	No	2.88

the proposed LTS approach, when considering firstly MNIST and afterwards MNIST-Fashion, are presented in Fig. 6, considering 1,000 labeled images from the MNIST database and 10,000 labeled images from MNIST-Fashion. From these results we observe that although only a small part of labeled training data is available, the proposed approach preserves the performance achieved on the previous database while learning a new task. Traditional semi-supervised learning approaches cannot deal with the lifelong learning setting due to the catastrophic forgetting challenge. These results demonstrate the effectiveness of the data replay on relieving catastrophic forgetting. In addition, we also compare our approach to the state of the art semi-supervised approaches on MNIST and the results are provided in Table 3. The results obtained by LTS are better or at least similar with those of other algorithms that do not perform under the lifelong learning framework.

7.4 The lifelong learning of multiple databases

We evaluate the performance of LTS when learning longer sequences of datasets. We consider the following databases, which contain classes with completely different images from each another: MNIST, CIFAR10, Sub-ImageNet and CelebA. Sub-ImageNet is created by randomly choosing 60,000 from the ImageNet database [56] for training, and 10,000 images

(a) Data samples (b) VAE results. (c) WGAN results.

Fig. 7. Generation and reconstruction results for LTS when considering unsupervised training with MNIST, CIFAR10, Sub-ImageNet and CelebA databases.

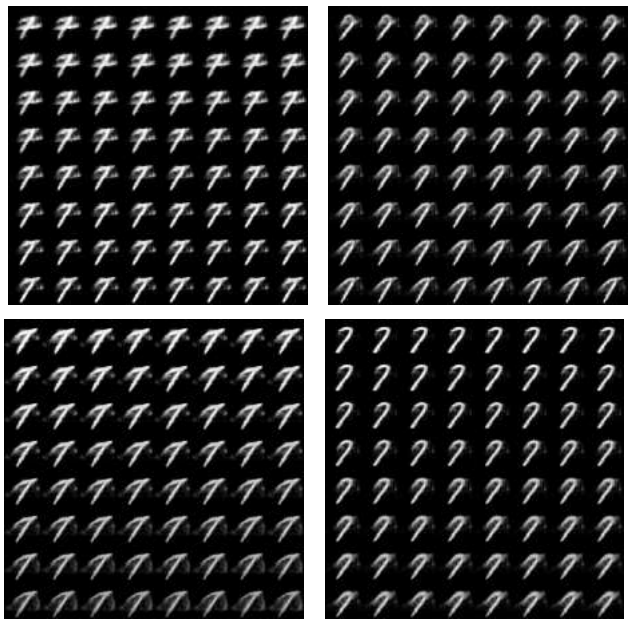


Fig. 8. Generation results in digit images after the LTS lifelong learning of MNIST and MNIST-Fashion database, when changing a single latent variable from -2 to 2.

for testing. We resize all images to $32 \times 32 \times 3$ pixels. We only consider two encoders for the unsupervised LTS training, as defined through the L_{VAE2} cost function from equation (28). The results for the average Negative Log-Likelihood (NLL) and the Inception Score (IS) [57], showing the quality of reconstructed images, are provided in Tables 4 and 5, respectively. Selected real images from the four databases are shown in Fig. 7a, while the results produced after learning all four databases are provided in Fig. 7b for the VAE Student network and in Fig. 7c for the WGAN Teacher network, for $\beta_1 = 1$ in (28) during the training

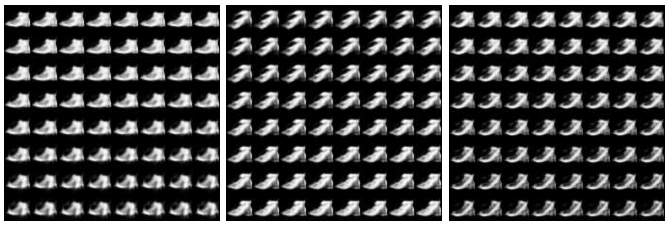


Fig. 9. Generation results in fashion item images after the LTS lifelong learning of MNIST and MNIST-Fashion database, when changing a single latent variable from -1 to 3.

(a) Real images.

(b) VAE Student Network reconstructions.

(c) Images generated by the WGAN Teacher network.

Fig. 10. Generation and reconstruction results following the unsupervised lifelong learning by the proposed approach on Celeba and CACD databases.

procedure. From these results we observe that both VAE Student and WGAN Teacher modules can reconstruct and generate images of high quality, even after training on a sequence of four completely different data sets.

In the following we consider the supervised learning setting, defined by L_{Stud} cost function from equation (18), where β_1, β_2 and β_3 are all set to 1. We train various models considering MNIST, SVHN and CIFAR10 databases. The classification accuracy evaluated on all testing samples is provided in Table 6. From the results provided in the Tables 4, 5 and 6 we can see that the proposed LTS method provides the best results when averaging the lifelong learning results on all databases considered.

TABLE 4
Average NLL on all testing samples after the lifelong learning of MNIST, CIFAR10, Sub-ImageNet, CelebA.

Database	LTS	CURL [58]	LGM [33]
MNIST	402.63	440.58	430.92
CIFAR10	255.23	283.68	620.57
Sub-ImageNet	243.10	282.14	458.60
CelebA	160.78	255.18	363.04
Average	265.43	315.39	468.28

TABLE 5
IS score on 5,000 testing data after the lifelong learning of MNIST, CIFAR10, Sub-ImageNet, CelebA.

Database	LTS	CURL [58]	LGM [33]
CIFAR10	3.97	3.53	3.46
Sub-ImageNet	4.00	3.60	3.55

TABLE 6
Average classification accuracy on all testing data after the lifelong learning of MNIST, SVHN and CIFAR10.

Database	L-TS	CURL [58]	LGM [33]	MemoryGANs [53]
MNIST	92.83	94.66	94.53	94.58
SVHN	67.93	33.53	31.23	66.72
CIFAR10	57.03	66.58	64.08	58.62
Average	72.60	64.92	63.61	61.34

Fig. 11. Interpolation results after the lifelong learning from CelebA to CACD. The original images are shown at the ends of each row and the interpolations are located in between. The first two rows show interpolations between the images from different domains while the last two rows show interpolations using images from the same database.

7.5 Supervised learning of disentangled representations

In this section, we evaluate the effectiveness of the proposed approach for supervised lifelong disentangled representation learning. We consider two distinct data sets, MNIST and MNIST-Fashion, where we also know the class labels and adopt the same LTS architecture as in Section 7.2. We train the LTS model using the Adam algorithm [59] and the objective function L_{Stud} from (18), considering $\beta_1 = 4$, $\beta_2 = 1$ and $\beta_3 = 1$ while training for a maximum number of 10 epochs for each learning phase considering a training rate of 0.001. We manipulate the generated images, by changing the learnt data attributes, after the LTS learning. The results, where we change a single continuous latent variable each time while fixing the others, for data from MNIST and MNIST-Fashion, are presented in Fig. 8 and Fig. 9, respectively. From these results we observe that the proposed approach is able to capture the thickness and the handwriting style from the images of digits from the MNIST database, while modelling the size and shape or various items from MNIST-Fashion. These results demonstrate that the proposed approach can capture both continuous and discrete data variations under the lifelong learning setting.

7.6 Unsupervised learning of disentangled representations

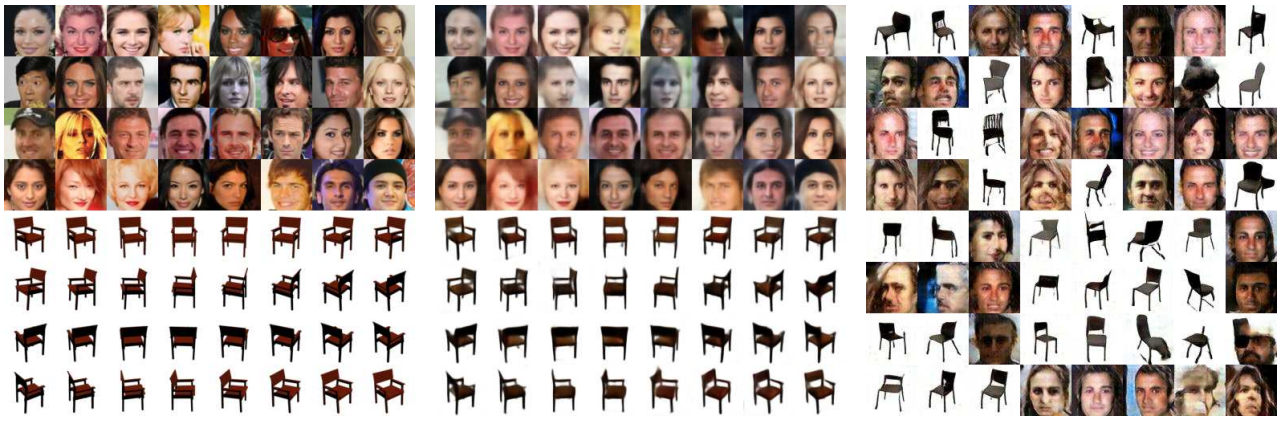
In this section, we test whether the proposed approach can learn disentangled representations under the lifelong unsupervised setting. We consider a deep CNN consisting of five convolution layers as the encoder, and the same number of layers for the decoder of the Student module. The number of filters in each layer is increased progressively with the depth of the network. Firstly, we evaluate the ability of the proposed approach to model complex data distributions. We consider two data sets showing human faces: CelebFaces Attributes data set (CelebA) [60] and Cross-Age Celebrity data set (CACD) [61]. CelebA contains more than 200K celebrity face images and each one has 40 attribute annotations. We use the random crop and resize for the images from CelebA, resulting in images of 64×64 pixels. CACD is also a large-scale celebrity face data set consisting of 163,446 images from 2,000 persons. We simply resize the CACD images to 64×64 pixels without considering cropping. We consider $\beta_1 = 1$ and $\beta_2 = 1$ in the loss function L_{VAE2} from (28). We train this model initially with images from CelebA and then with images from CACD database using the proposed Lifelong LTS framework. Real images are shown in Fig. 10a, while those reconstructed by the VAE Student and WGAN Teacher networks are shown in Figures 10b and 10c, respectively. From these results we observe that the proposed approach gives accurate reconstruction results although it does not use any real images from CelebA when being trained on the second database, CACD. In order to explore the joint latent spaces corresponding to CelebA and CACD databases, we perform interpolation experiments on these two different domains (Lifelong Learning Interpolation [18]). We randomly select two images, one from CACD and another from CelebA database, and interpolate between their corresponding latent spaces and then we map the resulting latent spaces back into the image domain. The interpolation results are evaluated on four pairs of images, chosen from the same and from different domains, respectively. The interpolation results are shown in Fig. 11, where the original images are shown at the ends of each row of images and those resulting from the latent space interpolations are displayed in between them. It can be observed that the interpolated images are smoothly transformed between each pair of original images, even when the source and target face images correspond to different image categories. These results show that the proposed approach can learn meaningful latent representations across multiple domains under the unsupervised lifelong learning setting.

We also consider the lifelong LTS learning of two databases with entirely different types of images: CelebA followed by the 3D-Chairs database, which displays a variety of 3D representations of chairs. Real images from CelebA are shown in the top 4 rows from Fig. 12a while the bottom 4 rows shows selected images from the 3D-Chairs database. After the lifelong learning of the probabilistic representations of these databases, in Figures 12b and 12c we show the reconstructions by VAE Student network and by the WGAN Teacher network, respectively, when considering $\beta_1 = 1$, $\beta_2 = 1$ in the loss function L_{VAE2} from (28). From these results it can be observed that the proposed LTS framework is able to provide good reconstructions in both databases. Then, in the last example we show interpolation experiments on pairs of images, where each is drawn from a different database as well as from the same database. Each of the results is shown on a row of images from Fig. 13, where the original images are at the ends of each row. From these results it can be observed that a chair is smoothly transformed into a human face in the first four rows from Fig. 13, when varying the interpolation weights in the latent space. We can observe that the main body of a chair is transformed into either the hair or the glasses worn by human subjects. We also observe that the interpolation results are smoothly changing when the original images are from the same database, either CelebA or 3D-chairs, as it can be seen in the results from the bottom two rows of Fig. 13.

In the following we train the LTS model with $\beta_1 = 4$, and $\beta_2 = 1$ in (28) when considering Lifelong LTS learning from CelebA to 3D-Chairs databases. After training, we modify one of the latent variables while fixing the others. The disentangled results on CelebA human faces and 3D-Chairs are shown in Figures 14 and 15, respectively. From these results it can be observed that the LTS model is able to discover disentangled representations for various data attributes, including skin color, gender, hair colour and baldness/hair variation in human face images as well as the size and colour of chairs in the 3D-Chairs' images.

7.7 Ablation study

In this section, we firstly consider a baseline, named LTS*, which does not optimize L_δ from (23), characterizing the training of the domain-specific encoder. We also consider a baseline that does not use the conditional prior characterizing the domain-specific generative factor from equation (21), and name this model as LTS**. Thus, the Student model in either LTS* or LTS** drops one of the encoders and uses only the two other encoders. We train all these models under the



(a) Real images from CelebA and 3D-chairs. (b) VAE Student network reconstructions. (c) WGAN Teacher network results.

Fig. 12. Image generation and reconstruction following the unsupervised lifelong LTS learning on CelebA to 3D-chairs.

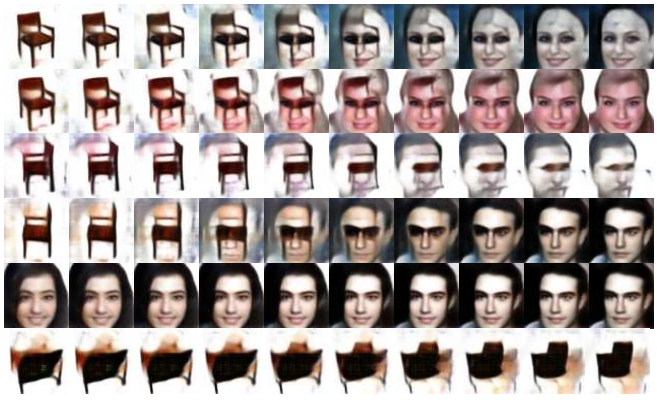


Fig. 13. Interpolation results following the LTS lifelong learning from CelebA to 3D-Chairs. In the first four rows, we interpolate two images from two different databases, shown at the ends of each row, while in the last two rows the chosen images are from the same domain, CelebA and 3D-Chairs. The interpolated generated images are shown in between the real images.

lifelong learning of MNIST, CIFAR10, Sub-ImageNet (Sub-I) and CelebA and the results are provided in Table 7. The results from this table indicate the crucial role played by the domain-specific encoder in the Student module, trained by (23), and the conditional prior characterizing the domain-specific generative factor from (21). This result also shows that the performance of the LTS model is improved by embedding information from different domains into several distinct clusters in the latent space.

TABLE 7
The average Negative log-likelihood (NLL) on all testing data samples after the lifelong learning of MNIST, CIFAR10, Sub-ImageNet, CelebA.

Methods	MNIST	CIFAR10	Sub-I	CelebA	Average
LTS	402.63	255.23	243.10	160.78	265.43
LTS*	504.92	309.93	309.78	279.75	351.09
LTS**	261.16	511.18	466.46	251.49	372.57

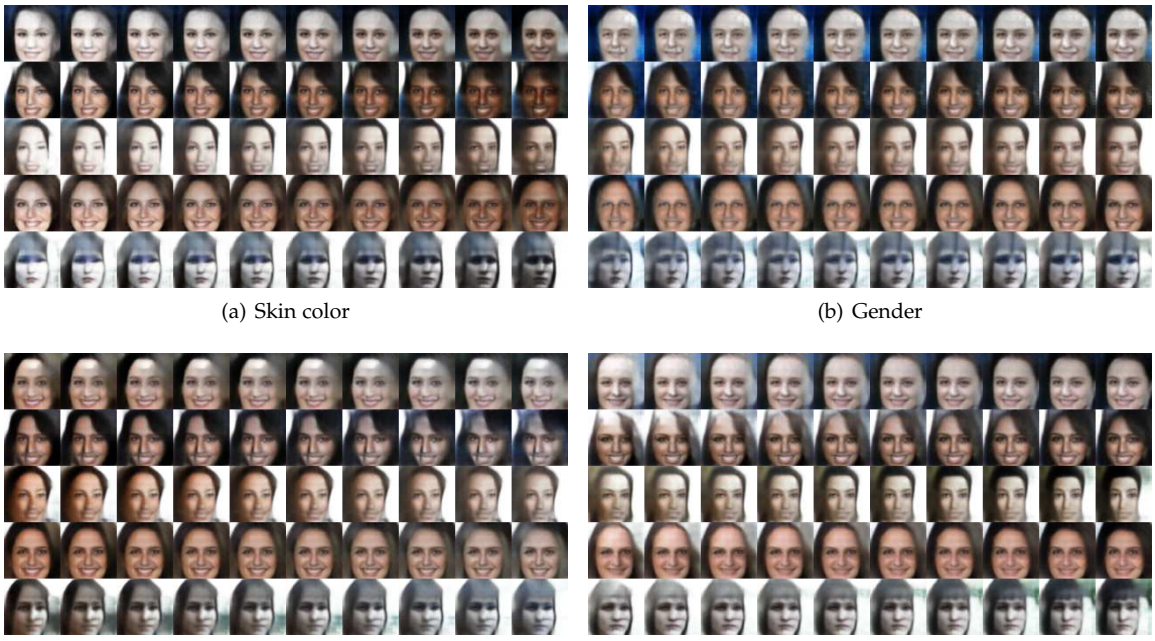
We also consider a baseline model that does not optimize the loss function defining discrete variables through the specific encoder of the Student module $L_{\mathbf{s}}$, defined by equation (22), and we train the resulting model under the supervised learning setting. The forgetting curve, evaluating the classification accuracy, is provided in Fig. 16, where we can observe that the baseline without the supervised loss can not predict accurate labels for the given data samples.

7.8 Discussion

In the following, we evaluate the error bounds for the lifelong learning of the Student module, derived according to the study from Section 6. We consider the proposed model when jointly training with the MNIST and SVHN, both databases representing images of digits, and call this as LTS Joint Distribution Training (LTS-JDT). We also consider the lifelong learning using the LTS model, of these databases, considering 20 epochs for training with each task. We evaluate the average risks on all testing data for the Student module for LTS-JDT, implemented by a VAE, using Definition 3. This model can be seen as the Teacher which approximates the joint distribution $\widehat{\mathcal{D}}^i$ when learning each i-th task while the Student module is trained on the true joint distribution $\widetilde{\mathcal{D}}^{1:2}$. From equation (29) we have $\sum_{i=1}^{2} \mathrm{R}_{\mathcal{D}^i_{\mathcal{X}}}(h, f)$, where h represents a mapping $h : \mathcal{X} \to \mathcal{Y}$, corresponding to $p_{\theta_3}(\mathbf{s}|\mathbf{x})$, the component in the Student's objective function L_{Stud} from (18), inferring the class label, and $f : \mathcal{X} \to \mathcal{Y}$ represents the true labeling function. We also train the LTS model during the lifelong learning using successively MNIST and SVHN while evaluating the risk on the target datasets for the Student module. The results are provided in Fig. 17. We observe that if the Teacher does not approximate the joint distribution exactly in each task learning, the performance of the Student degenerates when learning more tasks.

In the following, we evaluate the forgetting rate for the information learnt from the first database by the Student module while learning a long sequence of tasks. The NLL results, evaluated on MNIST data, when engaging in the lifelong learning of MNIST, CIFAR10, Sub-ImageNet and CelebA databases, are provided in Fig. 18. These databases contain very different categories of images and while the images from the first database are simple, the other databases contain complex images. The results from Fig. 18 indicate that the Student component of the LTS model tends to have higher errors as it learns additional tasks.

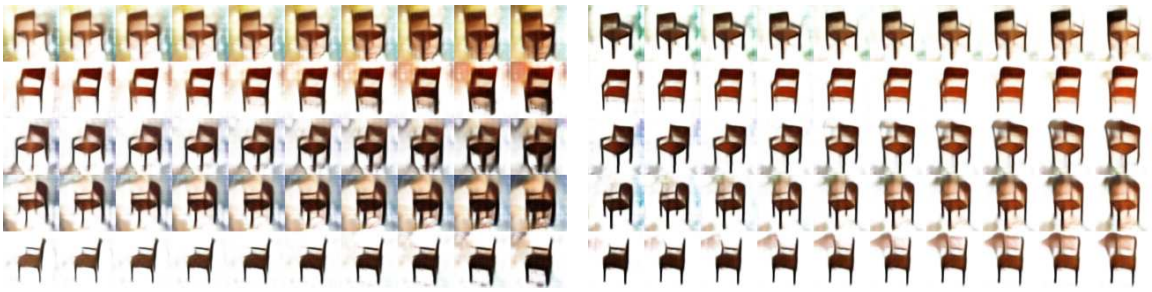
The Teacher module is required to refine, process and preserve previously learnt knowledge. However, the quality



(a) Skin color (b) Gender

(c) Hair color (d) Baldness/hair

Fig. 14. Results of attribute manipulation in the generated images after learning the probabilistic representation for CelebA dataset under the Lifelong training from CelebA to 3D-Chairs. We change a single latent variables in the latent space from -3.0 to 3.0 while fixing the others.



(a) Size (b) Color

Fig. 15. Results of attribute manipulation in the generated images after learning the probabilistic representation for 3D-Chairs database under the Lifelong learning from CelebA to 3D-Chairs. We change a single latent variable in the latent space from -1.0 to 5.0 while fixing the others.

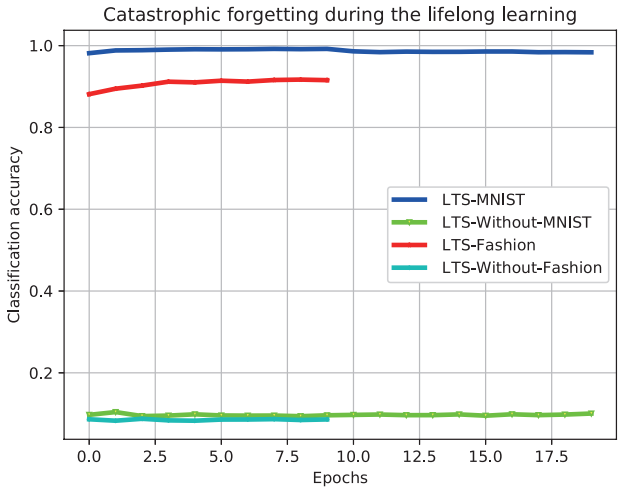


Fig. 16. Forgetting analysis of the proposed model during the lifelong learning of MNIST to Fashion databases.

of the generated knowledge by the Teacher module degenerates when learning a large number of tasks. From Theorem 2 in Section 6, we know that the gap on risks (evaluated by the Student module) between the target distribution and the approximate distribution, generated by the Teacher module, depends on the discrepancy distance Δ, from Definition 4. While GANs have very good generalization properties, they also have physical bounds in their information learning capacity. Therefore, the Teacher is not able to generate high-quality knowledge following the training with a long sequence of tasks. This problem is related to the mode collapse [62], and catastrophic forgetting [63] in GANs, where the discriminator constraints the ability to generate data corresponding to a diversity of modes in the given data. Consequently, the Student module, learning from the Teacher, is only able to capture a limited number of modes of variation across all the given tasks.

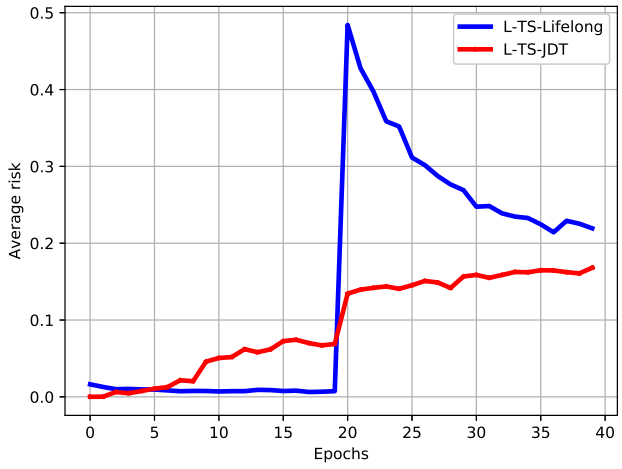


Fig. 17. Classification error curves when learning MNIST and SVHN databases, evaluating the testing data from both databases during training. LTS-Lifelong represents the lifelong learning curve when training from MNIST to SVHN database. LTS-JDT represents the results when training the LTS model directly with both databases.

8 Conclusions

We propose a novel lifelong deep learning approach by using a Teacher-Student framework for learning successively the probabilistic representations of a sequence of databases. The proposed framework consists of two components : a Teacher module implemented by a Wasserstein GAN which is used to generate the knowledge from all previously learnt databases, and a Student module, implemented by a VAE which is trained to capture both discrete and continuous meaningful variations across multiple domains. The VAE Student network is trained using the joint knowledge generated by the WGAN Teacher network for the previously learnt databases, and the current task, defined by a newly available database. The proposed framework is extended for three different learning situations: supervised, semi-supervised and unsupervised. Furthermore, the experimental results show that the proposed approach is able to discover disentangled and interpretable representations of multiple domains in an unsupervised lifelong learning setting. This study can lead to further research into how to accelerate the learning of future tasks as well as for evaluating the forgetfulness in artificial learning systems.

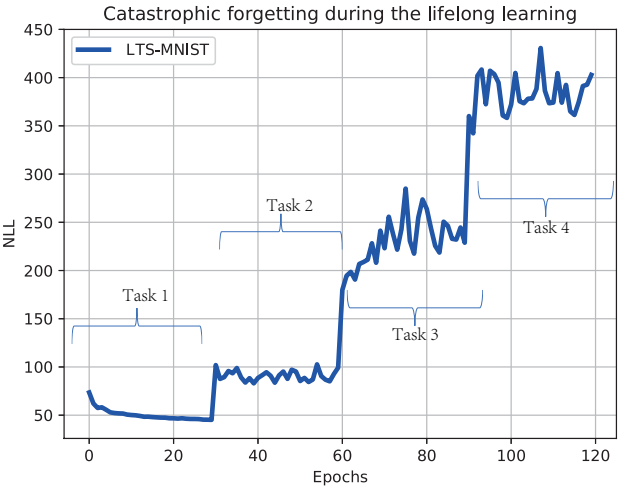


Fig. 18. Average NLL, calculated on the MNIST testing data samples, during the lifelong learning by the student VAE network of MNIST, CIFAR10, Sub-ImageNet and CelebA datasets.

References

[1] E. Akagündüz, A. G. Bors, and K. Evans, "Defining image memorability using visual memory schema," *IEEE Trans. on Pattern Anal. and Machine Intelligence*, vol. 42, no. 9, pp. 2165–2178, 2020.

[2] J. Fagot and R. G. Cook, "Evidence for large long-term memory capacities in baboons and pigeons and its implications for learning and the evolution of cognition," *Proc. of the National Academy of Sciences (PNAS)*, vol. 103, no. 46, pp. 17 564–17 567, 2006.

[3] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 119, pp. 54–71, 2019.

[4] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "β-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. Int. Conf. on Learning Representations (ICLR)*, 2017.

[5] R. Aljundi, P. Chakravarty, and T. Tuytelaars, "Expert gate: Lifelong learning with a network of experts," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recog. (CVPR)*, 2017, pp. 3366–3375.

[6] B. Ren, H. Wang, J. Li, and H. Gao, "Life-long learning based on dynamic combination model," *Applied Soft Computing*, vol. 56, pp. 398–404, 2017.

[7] R. Polikar, L. Upda, S. S. Upda, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Transactions on Systems Man and Cybernetics Part C*, vol. 31, no. 4, pp. 497–508, 2001.

[8] G. M. van de Ven and A. S. Tolias, "Three scenarios for continual learning," in *NeurIPS Continual Learning workshop*, 2019. [Online]. Available: https://arxiv.org/abs/1904.07734

[9] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[10] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: a survey of recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 53–69, 2015.

[11] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proc. of Int. Conf. on Machine Learning (ICML)*, 2011, pp. 513–520.

[12] W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," Tech. Rep., 2018. [Online]. Available: https://arxiv.org/abs/1812.11806

[13] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[14] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2016, pp. 2172–2180.

[15] T. Q. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud, "Isolating sources of disentanglement in variational autoencoders," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2018, pp. 2615–2625.

[16] S. Gao, R. Brekelmans, G. V. Steeg, and A. Galstyan, "Auto-encoding total correlation explanation," in *Proc. Int. Artif. Intell. and Statistics (AISTATS), vol. PMLR 89*, 2019, pp. 1157–1166.

[17] H. Kim and A. Mnih, "Disentangling by factorising," in *Proc. Int. Conf. on Machine Learning, vol. PMLR 80*, 2018, pp. 2649–2658.

[18] A. Achille, T. Eccles, L. Matthey, C. Burgess, N. Watters, A. Lerchner, and I. Higgins, "Life-long disentangled representation learning with cross-domain latent homologies," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2018, pp. 9873–9883.

[19] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[20] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *NIPS Deep Learning and Representation Learning Workshop*, 2014. [Online]. Available: https://arxiv.org/abs/1503.02531

[21] H. Jung, J. Ju, M. Jung, and J. Kim, "Less-forgetting learning in deep neural networks," 2016. [Online]. Available: https://arxiv.org/abs/1607.00122

[22] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proc. of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[23] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *Proc. of Int Conf. on Machine Learning (ICML), vol. PMLR 70*, 2017, pp. 3987–3995.

[24] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proc. Int. Conf. on Machine Learning (ICML)*, 2007, pp. 193–200.

[25] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," 2016. [Online]. Available: https://arxiv.org/abs/1606.04671

[26] G. Zhou, K. Sohn, and H. Lee, "Online incremental feature learning with denoising autoencoders," in *Proc. Int. Conf. on Artificial Intel. and Statistics (AISTATS), vol. PMLR 22*, 2012, pp. 1453–1461.

[27] C. Cortes, X. Gonzalvo, V. Kuznetsov, M. Mohri, and S. Yang, "AdaNet: Adaptive structural learning of artificial neural networks," in *Proc. of Int. Conf. on Machine Learning (ICML), vol. PMLR 70*, 2017, pp. 874–883.

[28] T. Xiao, J. Zhang, K. Yang, Y. Peng, and Z. Zhang, "Error-driven incremental learning in deep convolutional neural network for large-scale image classification," in *Proc. of ACM Int. Conf. on Multimedia*, 2014, pp. 177–186.

[29] J. L. Part and O. Lemon, "Incremental online learning of objects for robots operating in real environments," in *Proc. Joint IEEE Int. Conf. on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, 2017, pp. 304–310.

[30] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013. [Online]. Available: https://arxiv.org/abs/1312.6114

[31] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 32(2)*, 2014, pp. 1278–1286.

[32] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 2990–2999.

[33] J. Ramapuram, M. Gregorova, and A. Kalousis, "Lifelong generative modeling," *Neurocomputing*, vol. 404, pp. 381 – 400, 2020.

[34] A. Seff, A. Beatson, D. Suo, and H. Liu, "Continual learning in generative adversarial nets," 2017. [Online]. Available: https://arxiv.org/abs/1705.08395

[35] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "FitNets: Hints for thin deep nets," in *Proc. Int. Conf. on Learning Representations (ICLR)*, 2015. [Online]. Available: https://arxiv.org/abs/1412.6550

[36] G. Zhou, Y. Fan, R. Cui, W. Bian, X. Zhu, and K. Gai, "Rocket launching: A universal and efficient framework for training well-performing light net," in *Proc. AAAI Conf. on Artif. Intel.*, 2018, pp. 4580–4587.

[37] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2014, pp. 2672–2680.

[38] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 70*, 2017, pp. 214–223.

[39] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *Proc. Int. Conf. on Learning Representations (ICLR)*, 2018. [Online]. Available: https://arxiv.org/abs/1611.01144

[40] C. Maddison, D. Tarlow, and T. Minka, "A* sampling," in *Advances in Neural Inf. Processing Systems (NIPS)*, 2014, pp. 3086–3094.

[41] A. Creswell, Y. Mohamied, B. Sengupta, and A. A. Bharath, "Adversarial information factorization," 2018. [Online]. Available: https://arxiv.org/abs/1711.05175

[42] M. Ilse, J. M. Tomczak, C. Louizos, and M. Welling, "DIVA: Domain invariant variational autoencoders," in *ICLR Workshop DeepGenStruct*, 2019. [Online]. Available: https://arxiv.org/abs/1905.10427

[43] J. Klys, J. Snell, and R. Zemel, "Learning latent subspaces in variational autoencoders," in *Advances in Neural Inf. Processing Systems (NIPS)*, 2018, pp. 6445–6455.

[44] S. Narayanaswamy, T. B. Paige, J.-W. Van de Meent, A. Desmaison, N. Goodman, P. Kohli, F. Wood, and P. Torr, "Learning disentangled representations with semi-supervised deep generative models," in *Advances in Neural Inf. Processing Systems (NIPS)*, 2017, pp. 5925–5935.

[45] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner, "Understanding disentangling in β-VAE," in *NIPS Work. on Learning Disentangled Representations*, 2017. [Online]. Available: https://arxiv.org/abs/1804.03599

[46] I. Redko, E. Morvant, A. Habrard, M. Sebban, and Y. Bennani, *Advances in Domain Adaptation Theory*. ISTE Press - Elsevier, 2019.

[47] ——, "A survey on domain adaptation theory: learning bounds and theoretical guarantees," 2020. [Online]. Available: https://arxiv.org/abs/2004.11829

[48] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation: Learning bounds and algorithms," in *Proc. Conf. on Learning Theory (COLT)*, 2009. [Online]. Available: https://arxiv.org/abs/0902.3430

[49] Y. Pu, W. Wang, R. Henao, C. L., Z. Gan, C. Li, and L. Carin, "Adversarial symmetric variational autoencoder," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2017, pp. 4333–4342.

[50] L. Mescheder, S. Nowozin, and A. Geiger, "Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks," in *Int. Conf. on Machine Learning (ICML), vol. PMLR 70*, 2017, pp. 2391–2400.

[51] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[52] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[53] C. Wu, L. Herranz, X. Liu, Y. Wang, J. van de Weijer, and B. Raducanu, "Memory replay GANs: Learning to generate new categories without forgetting," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2018, pp. 5962–5972.

[54] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," 2017. [Online]. Available: https://arxiv.org/abs/1708.07747

[55] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2014, pp. 3581–3589.

[56] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2012, pp. 1097–1105.

[57] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2016, pp. 2234–2242.

[58] D. Rao, F. Visin, A. Rusu, R. Pascanu, Y. W. Teh, and R. Hadsell, "Continual unsupervised representation learning," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2019, pp. 7645–7655.

[59] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. on Learning Representations (ICLR)*, 2015. [Online]. Available: https://arxiv.org/abs/1412.6980

[60] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. of IEEE Int. Conf. on Computer Vision (ICCV)*, 2015, pp. 3730–3738.

[61] B.-C. Chen, C.-S. Chen, and W. H. Hsu, "Cross-age reference coding for age-invariant face recognition and retrieval," in *Proc. European Conf. on Comp. Vision (ECCV), vol. LNCS 8694*, 2014, pp. 768–783.

[62] A. Srivastava, L. Valkov, C. Russell, M. U. Gutmann, and C. Sutton, "VEEGAN: Reducing mode collapse in GANs using implicit variational learning," in *Advances in Neural Inf. Proc. Systems (NIPS)*, 2017, pp. 3308–3318.

[63] H. Thanh-Tung and T. Tran, "Catastrophic forgetting and mode collapse in GANs," in *Proc. Int. Joint Conf. on Neural Networks (IJCNN)*, 2020, pp. 1–10.

Fei Ye is a currently third-year PHD student in computer science from University of York. He received the bachelor degree from Chengdu University of Technology, China, in 2014 and the master degree in computer science and technology from Southwest Jiaotong University, China, in 2018. His research topic includes deep generative image model, lifelong learning and mixture models.

Adrian G. Bors (Senior Member, IEEE) received the M.Sc. degree in electronics engineering from the Polytechnic University of Bucharest, Bucharest, Romania, in 1992, and the Ph.D. degree in informatics from the University of Thessaloniki, Thessaloniki, Greece, in 1999. In 1999, he joined the Department of Computer Science, University of York, York, U.K., where he is currently a Lecturer.

In 1999 he joined the Department of Computer Science, Univ. of York, U.K., where he is currently a lecturer. Dr. Bors was a Research Scientist at Tampere Univ. of Technology, Finland, a Visiting Scholar at the Univ. of California at San Diego (UCSD), and an Invited Professor at the Univ. of Montpellier, France. Dr. Bors has authored and co-authored more than 140 research papers including 32 in journals. His research interests include computer vision, computational intelligence and image processing.

Dr. Bors was a member of the organizing committees for IEEE WIFS 2021, IPTA 2020, IEEE ICIP 2018, BMVC 2016, IPTA 2014, CAIP 2013, and IEEE ICIP 2001. He was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING from 2010 to 2014 and the IEEE TRANSACTIONS ON NEURAL NETWORKS from 2001 to 2009. He was a Co-Guest Editor for a special issue on Machine Vision for the International Journal for Computer Vision in 2018 and the Journal of Pattern Recognition in 2015.